ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI DOTTORATO IN SCIENZE ECONOMICHE E STATISTICHE

DOTTORATO DI RICERCA IN

METODOLOGIA STATISTICA

PER LA RICERCA SCIENTIFICA

Ciclo XXVI

PERMUTATION-BASED STOCHASTIC ORDERING USING PAIRWISE COMPARISONS

Federico Mattiello

Dipartimento di Scienze Statistiche "Paolo Fortunati"

Bologna, October 31, 2014

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

SCUOLA DI DOTTORATO IN SCIENZE ECONOMICHE E STATISTICHE

DOTTORATO DI RICERCA IN

METODOLOGIA STATISTICA

PER LA RICERCA SCIENTIFICA

Ciclo XXVI

PERMUTATION-BASED STOCHASTIC ORDERING USING PAIRWISE COMPARISONS

Federico Mattiello

Coordinatrice Dottorato: *Prof.ssa* Angela Montanari
Relatore: *Prof.ssa* Rossella Miglio
Correlatore: *Prof.* Olivier Thas
Settore Concorsuale: 13/D1
Settore Disciplinare: SECS-S/01
Esame finale anno: 2015

Dipartimento di Scienze Statistiche "Paolo Fortunati"

Bologna, October 31, 2014

Federico Mattiello: *Permutation-Based Stochastic Ordering Using Pairwise Comparisons,* Tesi di Dottorato, © october 31, 2014.

"Discovery consists in seeing what everybody else has seen, and thinking what nobody else has thought"

— Albert Szent-Györgyi de Nagyrápolt

"I have the simplest tastes. I am always satisfied with the best."

— Oscar Wilde

"The ability to quote is a serviceable substitute for wit."

— W. Somerset Maugham

Per Teresina

SUMMARY

The topic of this work concerns nonparametric permutation-based methods aiming to find a ranking (stochastic ordering) of a given set of groups (populations), gathering together information from multiple variables under more than one experimental designs.

The problem of ranking populations arises in several fields of science from the need of comparing $G \geqslant 2$ given groups or treatments when the main goal is to find an order while taking into account several aspects.

As it can be imagined, this problem is not only of theoretical interest but it also has a recognised relevance in several fields, such as industrial experiments or behavioural sciences, and this is reflected by the vast literature on the topic, although sometimes the problem is associated with different keywords such as: "stochastic ordering", "ranking", "construction of composite indices" *etc.*, or even "ranking probabilities" outside of the strictly-speaking statistical literature.

The properties of the proposed method are empirically evaluated by means of an extensive simulation study, where several aspects of interest are let to vary within a reasonable practical range. These aspects comprise: sample size, number of variables, number of groups, and distribution of noise/error.

The flexibility of the approach lies mainly in the several available choices for the test-statistic and in the different types of experimental design that can be analysed. This render the method able to be tailored to the specific problem and the to nature of the data at hand.

To perform the analyses an `R` package called `SOUP` (**S**tochastic **O**rdering **U**sing **P**ermutation) has been written and it is available on `CRAN`.

ACKNOWLEDGMENTS

When writing a thank note it is always difficult to list all the people that helped in the process and I apologise in advance for all the omissions.

As far as the thesis work is concerned, I want to thank my supervisor Rossella Miglio as well as my co-supervisor Olivier Thas because without their support, feedbacks and suggestions I would not have been able to finish this work. A thank goes also to my previous PhD coordinator Daniela Cocchi and my current one Angela Montanari for they have shown tolerance and helpfulness along these four years I spent between Bologna and Gent.

I want to thank also the members of the examination committee for their useful suggestion and corrections, and in particular Livio Finos for the several permutation-related discussions we have had since I started working on these topics in 2009.

A special thank to Valentina that, despite all the practical difficulties we have faced together in the last three years, has endured them all, standing always by my side. She is the best partner and companion I could wish for.

Last but not least I want to thank my family, the support of which is always present and never taken for granted.

Gent & Este, october 31, 2014 $\mathcal{F}$. $\mathcal{M}$.

CONTENTS

LIST OF FIGURES

1 INTRODUCTION

The topic of this work concerns nonparametric permutation-based methods aiming to find a ranking (stochastic ordering) of a given set of groups (populations), gathering together information from multiple variables.

The problem of ranking populations arises from the need of comparing $G \geqslant 2$ given groups or treatments when the main goal is to find an order while taking into account several aspects. As it can be imagined, this problem is not only of theoretical interest but it also has a recognised relevance in several fields, such as industrial experiments or behavioural sciences, and this is reflected by the vast literature on the topic, although sometimes the problem is associated with different keywords such as: "stochastic ordering", "ranking", "construction of composite indices" *etc.*.

The properties of the proposed method are empirically evaluated by means of an extensive simulation study, where several aspects of interest are let to vary within a reasonable practical range. These aspects comprise: sample size, number of variables, number of groups, and distribution of noise/error. The flexibility of the approach lies mainly in the several available choices of test-statistic that can be used, this render the method able to be tailored to the specific problem and to the nature of the data at hand.

Very often, when ordering a set of groups or treatments the research aim is focused on evaluating their performances from a multivariate point of view, that is, in connection with more than one aspect (dimension, variables) and/or under several conditions (stratification).

The merging of information coming from several variables is performed through the *NPC* methodology (Pesarin and Salmaso,

2010b), that preserves the dependence structure among variables in a nonparametric fashion (*w.r.t.* the underlying distribution), and hence lets the experimenter free from modelling it. *Mahalanobis distance* was considered as a reference but it is worth noting that it can be applied only when the number of variables does not exceed the sample size whereas *NPC* works variable-by-variable and, under certain conditions, enjoys the property of finite-sample consistency (see Pesarin and Salmaso, 2010a).

Historically, the problems of statistical inference were basically formulated as those of estimation or testing of hypotheses. This formulation, however, does not exactly suit the objectives of an experimenter in many situations when he is faced with the problem of comparing several populations. These are generally the populations of the responses to certain "treatments". In all these problems, we have $G \geqslant 2$ groups and each population is characterized by the value of a parameter θ, which may denote, for example, the average of some meaningful variables for a variety of treatments. Indeed, in many practical settings, when comparing systems or groups, we are interested simultaneously in two or more characteristics of each component or individual, so our observations are vector-valued and the components of the vector observations may be referred to correlated random variables.

In these scenarios, the problem can be complicated and some methodological and practical issues arise: standardization, multivariate structure of data, accuracy of partial indicators, distance with respect to a target, stratification in presence of confounding factors *etc.*

The classical approach in all the preceding situations is to test the hypothesis of homogeneity of the parameters: $H_0 : \theta_1 = \cdots = \theta_G$ where $\theta_1, \ldots, \theta_G$ are the values of the parameter θ for these populations. In the general setting θs are functionals, i.e. functions of all parameters defining the involved distributions. With clear meaning of the symbols, $\theta_i = \int_{\Omega_i} \theta(X) dF_i(X)$, where $F_i(X)$ is the c.d.f. and Ω_i is the sample space of the i-

th group. If the populations are assumed to be normal with means $\theta_1, \ldots, \theta_k$, and a common variance σ^2, then the test can be carried out by using the one-way analysis of variance technique. In cases of other distributions for which θ may denote a different measure, one can develop a test of the null hypothesis H_0 using the Neyman-Pearson theory. Such tests have been developed for various situations and many of these are available in the statistical literature.

It should, however, be recognized that a satisfactory solution to any statistical inference problem depends on the goal of the experiment. In this sense, the classical tests of homogeneity cannot provide a satisfactory solution for these problems because the goal of the experimenter is to identify the variety with the largest average (most effective product, most effective educational system, most effective drug, and so on) rather than just to accept or reject the homogeneity hypothesis. Indeed, when the homogeneity test is carried out and its result is significant, the experimenter faces some real problems: he could use , for example, the method of least significant differences, based on t-tests, to detect differences between θs and thus to choose the "best" population. Nevertheless this method is at best indirect and less efficient, because it lacks protection in terms of a guaranteed probability against selecting a wrong population as "best".

Furthermore, although in general one may define a partial order relationship (reflexive, antisymmetric, and transitive relationship where not all elements are comparable) on the set of multivariate normal distributions, defining a real-valued function θ_i of the parameters μ_i (usual mean), and Σ_i (variance-covariance matrix), and use the θ_i to rank the populations, when distribution are substantially different, and particularly when the assumption of multivariate normal distribution is not realistic (quite often in practical situations), these function θ_i has to be constructed carefully and possibly using different methods.

The present work builds upon my Master Thesis (Mattiello, 2010) that has been already successfully applied in at least two situations: the main one was in the field of product benchmarking of industrial experiments regarding detergents, analyses performed on a daily basis (see Arboretti Giancristofaro et al., 2010); the second one was comparing the effectiveness of two *media* on a target audience by means of questionnaires (see Mattiello et al., 2012).

1.1 LITERATURE REVIEW

Looking at the literature, under the keywords "Stochastic Order" or "Ranking", few but extensive works can be found. For example, the book of Gupta and Panchapakesan (2002) presents various approaches to ordering and ranking several populations under different approaches, here the problem of multiple comparisons related to ranking is treated from a theoretical point of view and lots of conditions and properties are examinated.

Shaked and Shanthikumar (2007), instead, describes and analyses the properties of several concepts of univariate and multivariate stochastic orders and orderings while still keeping a highly theoretical level.

These works treat a great number of procedures focusing on theoretical aspects such as:

- defining ranking rules that respect a given probability of correct selection
- providing formulas for choosing the minimum sample size such that this probability is attained,
- selecting the best subset within a population
- comparing the differences between selecting only the best group or the best together with the second-best group,

- defining and studying the properties of several concepts of *stochastic order* from the univariate and multivariate point of view.

They do not provide, though, practical algorithms that can be directly applied in real situations where the sample size has been already fixed, like observational studies, and they deal with the problem of multiple comparisons by asking the experimenter to define beforehand a meaningful indicator (test-statistic) that will heavily influence the subsequent inference. For example, there is no clear indication on how to deal with information from pairwise multiple comparisons especially from the multivariate point of view.

The approach presented in this work falls under the same framework of Basso et al. (2009) but instead of analysing the properties of several test statistics and stochastic orders we will try to give a kind of a general algorithm that can be used with several test statistics and when the main research goal is to order the treatments or groups from the "best" to the "worst". The reader interested in this topic and framework can surely benefit also from the reading of Finos et al. (2008, 2007).

The main contribution of this work is contained in the third chapter where the main algorithm is described. Pairwise comparisons between groups are here exploited in such a way that, after the NonParametric Combination has been applied, it is possible to build a square matrix filled with p-values that can be used to test the correspondent pairwise hypotheses about the relative ordering of each group *w.r.t.* to the others. Subsequently, these rejections/non-rejections are used to come up with the final estimated ordering sequence, by means of a ranking rule stemming from *round-robin* sport tournaments.

1.2 ON THE CHAPTERS THAT FOLLOW

The present thesis work is organized as follows:

THE SECOND CHAPTER is a brief review on the basic aspects and concepts of univariate and multivariate permutation tests.

IN THE THIRD CHAPTER the core algorithm is described together with the various available features.

THE FOURTH CHAPTER is devoted to the simulation study in which the method is empirically tested under various conditions and different settings.

THE FIFTH CHAPTER presents a real case study coming from the field of early-stage drug development, it is a microarry experiment in which the expression level of several genes is measured under different conditions (administeration of several drugs/compounds).

CHAPTER SIX discusses the main features of the proposed method and suggests some further developments.

2 THE PERMUTATION APPROACH

Most of univariate statistical problems can be effectively solved with standard parametric or nonparametric methods although, in relatively mild conditions, their permutation counterparts are generally asymptotically as good as the best parametric ones. Moreover, there are a number of parametric tests the distributional behavior of which is only known asymptotically, while permutation methods are essentially of a nonparametrically exact nature in a conditional context. Thus, for most sample sizes in real applications, the relative lack of efficiency of permutation solutions may be sometimes compensated by the lack of approximation of parametric asymptotic counterparts.

Another aspect that is worth to point out is that even when responses are normally distributed, if there are too many nuisance parameters to estimate, the consequent reduction of degrees of freedom can makes the estimates inaccurate and so making preferable the permutation solution. Moreover, parametric methods often require some assumptions that are satisfied (and/or that is possible to check) in a very little number of applicational situations such as: homoscedasticity, normality and random sampling. This can lead the consequent inferences to be, when not improper, necessarily approximated.

In fact, parametric methods involve a modeling approach and generally require for a set of stringent assumptions, which are often quite unrealistic, unjustified or even too much *ad hoc* for specific inferential analysis, so that they may be more connected with the availability of methods than with reality. On the other hand, nonparametric approaches try to keep as less assumptions as possible, avoiding those which are difficult to justify or to interpret, and possibly without excessive loss of inferential efficiency. Thus, they are based on more realistic

foundations for statistical inference especially of observational studies and in general when experimental conditions are not systematically designed or clearly known.

This is not to claim the superiority of one approach over the other (that would be indefensible), but just to address some features of the parametric approach that makes the permutation (and conditional) approach more desirable in some situations. It seems much more reasonable to consider the two approaches *complementary* rather than mutually exclusive; this is why permutations approaches should be part of the toolbox of any statistician.

On one side, a wide range of inferential problems can be correctly and effectively solved within a permutation framework. Many complex multivariate problems that are common in areas such as: agriculture, biology, clinical trials, experimental design, genetics, pharmacology, psychology, and quality control, are difficult or even impossible to face outside the conditional context and in particular outside the Nonparametric Combination Methodology.

On the other side though, there are very important families of inferential problems which cannot be dealt with and/or solved in a permutation framework; some examples are: unconditional parametric estimation and testing, nonparametric kernel estimation, problems within statistical decision approach, and even most of the modelling frameworks. In addition, the traditional Bayesian inference also lies outside the permutation approach.

The key assumption that marks the applicability or not of the permutation approach is the *exchangeability condition*, at least under the null hypothesis, that will be described in the next section. But also in case too few observations are available, the permutation approach suffers from heavy approximations, although some power can be retrieved by switching to *e.g.* rotations instead of permutations (Langsrud, 2005), at the cost of losing the property of distribution invariance.

Sections that follow, and definitions therein, are summarised parts of the book Pesarin and Salmaso (2010b) (where these concepts are treated in a much deeper manner), functional to the problem being described in this work, and they are needed to give a theoretical background to the main algorithm.

2.1 EXCHANGEABILITY

For most problems of hypothesis testing, the observed data set $\mathbf{x} = \{x_1, \ldots, x_n\}$ is usually obtained by a symbolic experiment performed n times on a population variable X, which takes values in the sample space $\mathcal{X}$. Of course, when a data set is observed at its $\mathbf{x}$ value, it is presumed that a sampling experiment on a given underlying population has already been performed and the resulting sampling distribution is related to the parent population distribution, which is often denoted by P (see , for a deeper introduction).

Usually for the aim of analysis, the data set $\mathbf{x}$ is generally partitioned into *groups* or *samples*, according to the so-called *treatment levels* of the experiment.

Note that the observed data set $\mathbf{x}$ is always a set of sufficient statistics in H_0 for whatever underlying distribution. To see this let us assume that H_0 is true and all members of a nonparametric family $\mathcal{P}$ of non-degenerate and distinct distributions are dominated by one *dominating* measure "ξ". For a general testing problem, in the null hypothesis (H_0), which usually assumes that data come from only one (with respect to groups) unknown population distribution P, the whole set of observed data $\mathbf{x}$ is considered to be a random sample, taking values on a sample space $\mathcal{X}^n$, where $\mathbf{x}$ is one observation of the n-dimensional sampling variable $\mathbf{X}^{(n)}$ and where this random sample does not necessarily have independent and identically distributed (i.i.d.) components. Furthermore, let us denote by f_P the density of P with respect to ξ, by $f_P^{(n)}(\mathbf{x})$ the density of the sampling variable $\mathbf{X}^{(n)}$, and by $\mathbf{x}$ the data set. As the iden-

tity $f_P^{(n)}(\mathbf{x}) = f_P^{(n)}(\mathbf{x}) \cdot 1$ is true for all $\mathbf{x} \in \mathcal{X}^n$, except for points such that $f_P^{(n)}(\mathbf{x}) = 0$, due to the well-known factorization theorem any data set $\mathbf{x}$ is therefore a sufficient set of statistics for whatever $P \in \mathcal{P}$.

Definition 2.1.1 (Nonparametric Family of Distributions)**.** *A family of distributions $\mathcal{P}$ is said to be* nonparametric *when the parameter θ, belongs to a non-finite-dimensional parameter space Θ, such that there is not a one-to-one relationship between Θ and $\mathcal{P}$, in the sense that each member of $\mathcal{P}$ can not be identified by only one member of Θ, and* vice versa.

This definition includes families of distributions which are either unspecified or specified except for an infinite number of unknown parameters. All nonparametric families $\mathcal{P}$ which are of interest in permutation analysis are assumed to be sufficiently *rich* in a way that, if x and x' are any two points of $\mathcal{X}$, then $x \neq x'$ implies $f_P(x) \neq f_P(x')$ for at least one $P \in \mathcal{P}$, except for points with null density. Note that the fact that the family $\mathcal{P}$ is said to be nonparametric essentially depends on the knowledge we assume about it. For example, when we assume that the underlying family $\mathcal{P}$ contains all continuous distributions, then the data set $\mathbf{x}$ is *complete minimal sufficient*.

By *sufficiency, likelihood* and *conditionality principles of inference*, given a sample point $\mathbf{x}$, if $\mathbf{x}^* \in \mathcal{X}^n$ is such that the likelihood ratio
$f_P^{(n)}(\mathbf{x})/f_P^{(n)}(\mathbf{x}^*) = \rho(\mathbf{x}, \mathbf{x}^*)$ is not dependent on f_P for whatever $P \in \mathcal{P}$, then $\mathbf{x}$ and $\mathbf{x}^*$ are said to *contain essentially the same amount of information with respect to* P, so that they are equivalent for inferential purposes. The set of points which are equivalent to $\mathbf{x}$, with respect to contained information, is called *the orbit associated with* $\mathbf{x}$, and hereafter will be denoted by $\mathcal{X}^n_{/\mathbf{x}}$, so that $\mathcal{X}^n_{/\mathbf{x}} = \{\mathbf{x}^* : \rho(\mathbf{x}, \mathbf{x}^*)$ *is* f_P*-independent*$\}$.

It should also be noted that, when data are obtained by random sampling with i.i.d. observations, so that $f_P^{(n)}(\mathbf{x}) = \prod_{i=1}^n f_P(x_i)$, then the orbit $\mathcal{X}^n_{/\mathbf{x}}$ associated with $\mathbf{x}$ contains all permutations of $\mathbf{x}$ and, in this framework, the likelihood ratio satisfies the equation $\rho(\mathbf{x}, \mathbf{x}^*) = 1$.

The same conclusion is reached if $f_P^{(n)}(\mathbf{x})$ is assumed to be invariant with respect to permutations of the arguments of $\mathbf{x}$, i.e. the elements $(x_1, \ldots, x_n)$. This happens when the assumption of independence for observable data is replaced by that of exchangeability.

Definition 2.1.2 (Exchangeability)**.** *Data are said to be* exchangeable *if* $f_P^{(n)}(x_1, \ldots, x_n) = f_P^{(n)}(x_{u_1^*}, \ldots, x_{u_n^*})$*, where* $(u_1^*, \ldots, u_n^*)$ *is any permutation of* $(1, \ldots, n)$*.*

Note that, in the context of permutation tests, this concept of exchangeability is often referred to as the *exchangeability of the observed data with respect to groups*. Orbits $\mathcal{X}^n_{/\mathbf{x}}$ are also called *permutation sample spaces* . It is important to note that orbits $\mathcal{X}^n_{/\mathbf{x}}$ associated with data sets $\mathbf{x} \in \mathcal{X}^n$ always contain a finite number of points, as n is finite.

Roughly speaking, permutation tests are conditional statistical procedures, where conditioning is with respect to the *orbit* $\mathcal{X}^n_{/\mathbf{x}}$ *associated with the observed data set* $\mathbf{x}$. Thus, $\mathcal{X}^n_{/\mathbf{x}}$ plays the role of *reference set for the conditional inference* (see Lehmann and Romano (2005)). In this way, in the null hypothesis and assuming exchangeability, the conditional probability distribution of a generic point $\mathbf{x}^\dagger \in \mathcal{X}^n_{/\mathbf{x}}$, for whatever underlying population distribution $P \in \mathcal{P}$, is

(2.1.1)

$$\Pr\left\{\mathbf{x}^* = \mathbf{x}^\dagger | \mathcal{X}^n_{/\mathbf{x}}\right\} = \frac{\sum_{\mathbf{x}^*=\mathbf{x}^\dagger} f_P^{(n)}(\mathbf{x}^*) \cdot d\xi^n}{\sum_{\mathbf{x}^* \in \mathcal{X}^n_{/\mathbf{x}}} f_P^{(n)}(\mathbf{x}^*) \cdot d\xi^n} = \frac{\#\left[\mathbf{x}^* = \mathbf{x}^\dagger,\, \mathbf{x}^* \in \mathcal{X}^n_{/\mathbf{x}}\right]}{\#\left[\mathbf{x}^* \in \mathcal{X}^n_{/\mathbf{x}}\right]},$$

which is P-independent. Of course, if there is only one point in $\mathcal{X}^n_{/\mathbf{x}}$ whose coordinates coincide with those of $\mathbf{x}^\dagger$, i.e. if there are no ties in the data set, and if permutations correspond to permutations of the arguments, then this conditional probability becomes $1/n!$. Thus, $\Pr\left\{\mathbf{x}^* = \mathbf{x}^\dagger | \mathcal{X}^n_{/\mathbf{x}}\right\}$ is uniform on $\mathcal{X}^n_{/\mathbf{x}}$ for all $P \in \mathcal{P}$.

These statements allow permutation inferences to be invariant with respect to P in H_0. Some authors, emphasizing this

invariance property of permutation distribution in H_0, prefer to give them the name of *invariant tests*. However, due to this invariance property, permutation tests are distribution-free and nonparametric.

As a consequence, in the alternative hypothesis H_1, conditional probability shows quite different behavior and in particular it may depend on P. In order to illustrate this in a simple way, let us consider, for instance, a two-sample problem where $f_{P_1}^{(n_1)}$ and $f_{P_2}^{(n_2)}$ are the densities,relative to the same dominating measure ξ, of two sampling distributions related to two populations P_1 and P_2, which are assumed to differ at least in a set of positive probability. Suppose also that $\mathbf{x}_1$ and $\mathbf{x}_2$ are the two separate and independent data sets with sample sizes n_1 and n_2, respectively. Hence, the likelihood associated with the pooled data set is $f_P^{(n)}(\mathbf{x}) = f_{P_1}^{(n_1)}(\mathbf{x}_1) \cdot f_{P_2}^{(n_2)}(\mathbf{x}_2)$, and from the sufficiency principle it follows that the data set partitioned into two groups, $(\mathbf{x}_1; \mathbf{x}_2)$, is now the set of sufficient statistics. In fact, by joint invariance of the likelihood ratio with respect to both f_{P_1} and f_{P_2}, the orbit of $\mathbf{x}$ is $(\mathcal{X}_{/\mathbf{x}_1}^{n_1}, \mathcal{X}_{/\mathbf{x}_2}^{n_2})$, where $\mathcal{X}_{/\mathbf{x}_1}^{n_1}$ and $\mathcal{X}_{/\mathbf{x}_2}^{n_2}$ are partial orbits associated with $\mathbf{x}_1$ and $\mathbf{x}_2$, respectively. This implies that, conditionally, no datum from $\mathbf{x}_1$ can be exchanged with any other from $\mathbf{x}_2$ because in H_1 permutations are allowed only within groups, separately.

Consequently, when we are able to find statistics which are sensitive to the differences of two distributions, we can have a procedure for constructing permutation tests. Of course, when constructing permutation tests, one should also take into consideration the physical meaning of treatment effects, so that resulting inferential conclusions have clear interpretations.

Although the concept of conditioning for permutation tests is properly related to the formal conditioning with respect to orbit $\mathcal{X}_{/\mathbf{x}}^{n}$, hereafter we shall generally adopt a simplified expression for this concept by stating that *permutation tests are inferential procedures which are conditional with respect to the observed data set* $\mathbf{x}$. In fact, once $\mathbf{x}$ is known and the exchangeability condition is assumed in H_0, $\mathcal{X}_{/\mathbf{x}}^{n}$ remains completely determined by $\mathbf{x}$.

2.2 PERMUTATION TESTING PRINCIPLE

The act of conditioning on a set of sufficient statistics in H_0, and the assumption of exchangeability *with respect to groups* for observed data, make permutation tests independent of the underlying likelihood model related to P. As a consequence, P may be unknown or unspecified, either in some of its parameters or in its analytic form. This can be specified as follows:

Definition 2.2.1 (Permutation Testing Principle)**.** *If two experiments, taking values on the same sample space $\mathcal{X}^n$ and respectively with underlying distributions P_1 and P_2, both members of $\mathcal{P}$, give the same data set $\mathbf{x}$, then the two inferences conditional on $\mathbf{x}$ and obtained using the same test statistic must be the same, provided that the exchangeability of data with respect to groups is satisfied in the null hypothesis. Consequently, if two experiments, with underlying distributions P_1 and P_2, give respectively $\mathbf{x}_1$ and $\mathbf{x}_2$, and $\mathbf{x}_1 \neq \mathbf{x}_2$, then the two conditional inferences may be different.*

One of the most important features of the permutation testing principle occurs in multivariate problems, when solved through nonparametric combination methods . For this kind of problems, especially when they are complex and in very mild conditions, it is not necessary to specify or to model the structure of dependence relations for the variables in the underlying population distribution, so that analysis becomes feasible.

However, the conditioning on sufficient statistics provides permutation tests with good general properties. One of these is that, when exchangeability is satisfied in the null hypothesis, permutation tests are always exact procedures. One more and very important property is that their (non-randomized) conditional rejection regions are similar or α-invariant, in the sense of Scheffé (1943a), and Scheffé (1943b). The latter essentially means that if data come from continuous distributions, so that the probability of finding ties in the data set is zero, the rejection probability in H_0 is invariant with respect to observed data set $\mathbf{x}$, for almost all $\mathbf{x} \in \mathcal{X}^n$. Thus, conditional rejection regions are similar to the unconditional region. When

data come from non-continuous distributions, unless referring to randomized tests the similarity property is only asymptotically attained. Moreover, if the stochastic dominance condition is satisfied in H_1, permutation tests are *conditionally unbiased* procedures, since the rejection probability of any test T, for all data sets $\mathbf{x} \in \mathcal{X}^n$, satisfies the relation

$$\text{(2.2.1)}\quad \Pr\{\lambda(\mathbf{x}(\delta)) \leqslant \alpha | \mathbf{x}\} = W_\alpha(\mathbf{x}(\delta)) \geqslant \alpha$$

where $\lambda(\mathbf{x}(\delta))$ indicates the p-value and $W_\alpha(\mathbf{x}(\delta))$ the *conditional power* of T on $\mathbf{x}$ with fixed treatment effect δ and significance level α.

For this reason, permutation inferences are proper with observational data which sometimes are called non-experimental , and with well-designed sampling procedures. However, we must note that well-designed sampling procedures are quite rare even in most experimental problems. For instance, if we want to investigate the effects of a drug on rats, the units to be treated are usually not randomly chosen from the population of all rats, but are selected in some way among those available in a laboratory and are *randomly assigned* to the established treatment levels. The same occurs in most clinical trials, in which some of the patients present in a hospital are randomly assigned to one of the pre-established treatment levels.

In one sense, the concept of random sampling is rarely achieved in real applications because, for various reasons, real samples are commonly obtained by procedures affected by selection-bias. This implies that most of the forms of unconditional inferences usually associated with parametric tests, being based on the concept of random sampling, are rarely applicable in real situations. Additionally, due to the similarity and unbiasedness properties, permutation solutions allow for relaxation of most of the common assumptions needed by parametric counterparts, such as the existence of mean values and variances, and the homoscedasticity of responses in the alternative hypothesis.

Within the assumption of exchangeability in the null hypothesis, permutation conditional inferences always have a clear interpretation, whereas their weak unconditional extensions to the underlying parent population should be carried out and interpreted carefully. These extensions and associated interpretations are generally easy and correct when data are collected by well-designed random sampling techniques from a given population. Of course, if they are collected by selection-bias procedures, unconditional extensions may sometimes be ambiguous and misleading, although they are generally proper and correct for parent populations with respect to which actual data may be considered as a random sample.

Moreover, when exchangeability may be assumed in H_0, reference null distributions of permutation tests always exist, because (at least in principle), they are obtained by considering all permutations of available data. In addition, permutation comparisons of means do not require homoscedasticity in the alternative, provided that underlying c.d.f.s are ordered, so that they do not intersect each other. For these reasons, on the one hand, permutation inferences generally have a natural interpretation and, on the other, ordinary parametric tests are considered to be rarely applicable to real problems.

2.3 CONDITIONAL VS. UNCONDITIONAL INFERENCE

Unconditional parametric testing methods can be available, appropriate and effective when:

1. data-sets are obtained by well-defined random sampling procedures on well-specified parent populations;

2. population distributions (the likelihood models) for responses are well-defined;

3. with respect to all nuisance entities, well-defined likelihood models are provided with either boundedly complete estimates in H_0 or at least with invariant statistics;
4. at least asymptotically, null sampling distributions of suitable test statistics do not depend on any unknown entity.

Just as there are circumstances in which unconditional parametric testing procedures may be proper from the point of view of interpretation of related inferential results, so there are others in which they may become improper or even impossible. Conversely, there are circumstances in which conditional testing procedures may become appropriate and sometimes unique reasonable way. For example:

1. Distributional models are not well-specified.
2. Distributional models, although well-specified, depend on too many nuisance parameters.
3. Asymptotic null sampling distributions depend on unknown entities.
4. Sampling data come from finite populations or sample sizes are smaller than the number of parameters (or response variables).
5. In multivariate problems and in view of particular inferences, component variables have different degrees of importance.
6. Data sets are obtained by ill-specified selection-bias procedures
7. Treatment effects are presumed to act possibly on more than one aspect (a functional or pseudo-parameter), so that multi-aspect testing methods are of interest for inferential purposes.

Moreover, we could decide to adopt conditional testing inferences, not only when unconditional inference is not possible, but also when we are more interested in the actually observed data set than to the population model.

2.4 COMPUTATIONAL ASPECT

When sample sizes are not small, direct calculations are practically impossible because of the cardinality of associated permutation sample spaces $\mathcal{X}^n_{/\mathbf{x}}$ that quickly grows with n . Moreover, the approximation of such distributions by means of asymptotic arguments is not always appropriate, unless sample sizes are very large, because their dependence on $\mathbf{x}$ makes it difficult to express and to check the conditions needed to evaluate the degree of approximation in practical cases.

For practical reasons, in order to obtain appropriate and reliable evaluations of the permutation distributions involved is preferable to use a Monte Carlo procedures via *conditional simulation* for two reasons: 1) computing all possible permutations become rapidly very expensive from a computational point of view; 2) when drawing block diagrams of conditional simulation algorithms, one is forced to clarify the meaning and structure of all steps in the analysis. In addition, the usefulness of nonparametric combination methods is due to the fact that they allow to reduce the order of complexity of most analyses. In fact, they are applicable when a problem may be broken down into a set of partial sub-problems, each admitting a proper permutation solution.

It must be emphasized that conditional simulations are made using without-replacement resampling algorithms on the given data set, and so they are substantially different from the well-known *bootstrap techniques* that rely on the plug-in principle for the empirical cumulative distribution function.

In a *conditional simulation*, in order to consider random permutations, resampling replicates are done without-replacement on the given data set, considered as playing the role of a finite population. Hence, they correspond to a random sampling from the space of data permutations associated with the given data set. Of course, these conditional simulation procedures provide *statistical estimates* of desired permutation distributions, the accuracy of which depends on the number of replicates.

However, is important to remember that this kind of approximation is substantially closer to a statistical estimation than to a numerical evaluation.

2.5 PERMUTATION TEST: DEFINITION

In order to define a permutation test, let $\mathbf{X}$ be one data set of size n, $T : \mathcal{R}^n \to \mathcal{R}^1$ a test statistic, S all values of the related permutation support $\mathcal{T}(\mathbf{X}) = \left\{T^* = T(\mathbf{X}^*) : \mathbf{X}^* \in \mathcal{X}_{/\mathbf{X}}\right\}$ induced by $(T, \mathbf{X})$ and ordered in a non-decreasing way, $T^*_{(1)} \leqslant T^*_{(2)} \leqslant \ldots \leqslant T^*_{(S)}$; for any given $\alpha \in (0,1)$, $T_\alpha(\mathbf{X}) = T_\alpha = T^*_{(S_\alpha)}$ defines the permutation critical α-value associated with the pair $(T, \mathbf{X})$; the number $S_\alpha = \lfloor (1-\alpha) \cdot S \rfloor$, the integer part of $(1-\alpha) \cdot S$, defines how many permutation values T^* are less than the critical one.

Note that permutation critical values T_α depend on $\mathcal{X}_{/\mathbf{X}}$ and thus are defined assuming the null hypothesis is true with the data set $\mathbf{X}$. In other words, the critical values for a permutation test satisfy
$T_\alpha = T_\alpha(\mathbf{X}) = T_\alpha(\mathbf{X}^\dagger)$, $\forall\, \mathbf{X}^\dagger \in \mathcal{X}_{/\mathbf{X}}$, because the orbits of equivalent points associated with $\mathbf{X}$ and $\mathbf{X}^\dagger$ are such that $\mathcal{X}_{/\mathbf{X}} = \mathcal{X}_{/\mathbf{X}^\dagger}$. As a consequence, for all $\alpha \in (0,1)$, T_α are fixed values within support $\mathcal{T}(\mathbf{X})$, but they generally vary as $\mathbf{X}$, varies in $\mathcal{X}$.

Hence the structure of a (non-randomized) permutation testing (see Pesarin and Salmaso (2010b)) can be defined as follows:

$$\text{(2.5.1)}\quad \phi = \begin{cases} 1 & \text{if} \quad T^{obs} \geqslant T_\alpha, \\ 0 & \text{if} \quad T^{obs} < T_\alpha, \end{cases}$$

where T^{obs} represent the observed value of T on the dataset $\mathbf{X}$ and for which the associated type I error rate or *attainable* α-*value* are defined in the permutation support $\mathcal{S}_T = \{h/S,\ h = 1, 2, \ldots, S\}$ of $F_T(t|\mathbf{X})$, which is a discrete set whose cardinality depends on n, that is, not all possible desired values are available. We will hereafter refer to this kind of testing.

2.6 NONPARAMETRIC COMBINATION

The method of NonParametric Combination (or *NPC*) of a finite number of dependent permutation tests is a general tool for multivariate testing problems. It is useful when a set of quite mild conditions holds and when, as in many D-dimensional problems for continuous or categorical variables, one single appropriate overall test statistic $T \div \mathcal{R}^D \to \mathcal{R}^1$ is available.

In testing complex hypotheses, when the interest is on many different aspects, or when many response variables are involved, it is convenient first to process data using a finite set of $K > 1$ different *partial tests* (note that the number K of sub-problems is not necessarily equal to the dimensionality p of responses) that can be used in marginal or separate inferences, possibly after adjustment for multiplicity. On the other hand, if they are jointly considered they provide information on a global hypothesis, which is eventually the actual objective of most of the multivariate testing problems.

In the great number of situations it is unreasonable to assume a complete independence among these partial tests because they are functions of the same data set $\mathbf{X}$ and the component variables in $\mathbf{X}$ are generally not independent. Moreover, the underlying dependence relations among partial tests are rarely known except for some quite simple situations, and even when they are known they are often too difficult to cope with. Therefore, this combination must be done nonparametrically with respect to the underlying dependence relations, in this way there is no more the need to model the dependence relations among responses, aspect that is particularly relevant in many contexts.

2.6.1 Introduction to NPC

In order to illustrate the NPC [a] let us refer to the balanced one-way MANOVA layout, that is, we consider a D-dimensional

a hereafter we refer to NonParametric Combination in this way

data set
$\mathbf{Y} = \{\mathbf{Y}_k,\ k = 1,\dots,G\} = \left\{\mathbf{Y}_{jk},\ k = 1,\dots,G,\ j = 1,\dots,D\right\} =$
$= \left\{Y_{ijk},\ i = 1,\dots,G,\ j = 1,\dots,D,\ i = 1,\dots,n\right\}$ where the implicit assumption is that a family $\mathcal{P}$ of non-degenerate distributions exists, so the data set consists of $G \geqslant 2$ samples or groups of size n each, groups are presumed to be related to G levels of a treatment and $\mathbf{Y}_k$ are considered i.i.d. observations from $P_k \in \mathcal{P},\ \ k = 1,\dots,G$ (i.e. exchangeability may generally be enough, in place of independence). Of course this simple *balanced design* case can be complicated by the inclusion of covariates and/or by considering *unbalanced designs* (different sample sizes), but that is outside the scope of this introduction.

Let the (global) null hypothesis refer to equality of multivariate distributions of responses on G groups:

$$(2.6.1)\quad H_0 : \{P_1 = \cdots = P_G\} = \left\{\mathbf{Y}_1 \overset{d}{=} \dots \overset{d}{=} \mathbf{Y}_G\right\},$$

and let us suppose that either H_0 and H_1 may be properly and equivalently broken down into a finite set of sub-hypotheses H_{0c} and H_{1c}, $c = 1,\dots,C$, each appropriate for a partial aspect of interest. Therefore, H_0 is true if all the H_{0c} are jointly true, i.e. if $H_{0c} = \bigcap_{c=1}^{C} H_{0c}$ is true and, in the same way, H_0 is not true if at least one of the null sub-hypotheses H_{0c} is not true, i.e. $H_1 = \bigcup_{c=1}^{C} H_{1c}$. This imply that (under H_0) the D-variate data vectors of $\mathbf{Y}$ are exchangeable with respect to G groups; it is worth noting that in this work, as we are interested in determining locations from a multivariate point of view, C is equal to D.

2.6.2 Partial Tests Characterization

There are two principal side assumptions that are required to the set of partial test statistics $\mathbf{T} = \{T_c,\ c = 1,\dots,C\}$ and useful for nonparametric combination:

(i) all permutation partial tests T_c are marginally unbiased and significant for large values, so that they are stochastically larger in H_{1c} than in H_{0c}.

(ii) all permutation partial tests are consistent, i.e. $\Pr\{T_c \geqslant T_{c\alpha}|H_{1c}\} \to 1,\ \forall\alpha > 0,\ c = 1,\dots,C$, as $n \to \infty$, where $T_{c\alpha}$, is the α-level critical value of T_c and is assumed to be finite.

2.6.3 Combining Functions Characterization

A combining function is a function applied to p-values associated with partial tests, [b] that summarizes the information from these tests, nonparametrically with respect to the underlying dependence among terms. Hence the nonparametric combination of the second-order test is obtained with:

$$T' = \psi(\lambda_1,\dots,\lambda_C)$$

A generic combining function $\psi : (0,1)^C \to \mathcal{R}^1$, chosen from a class $\boldsymbol{\Psi}$, of combining function has to satisfy the following requirements:

1. A combining function ψ must be non-increasing in each argument: $\psi(\lambda_1\dots,\lambda_c,\dots\lambda_C) \geqslant \psi(\dots,\lambda'_c,\dots)$ if $\lambda_c < \lambda'_c$, $c \in \{1,\dots,C\}$.

2. ψ must be continuous in all C arguments, in that small variations in any subset of arguments imply a small variation in the ψ-index;

3. Every combining function ψ must attain its supremum value $\bar{\psi}$, possibly not finite, even when only one argument attains zero: $\psi(\dots,\lambda_c,\dots) \to \bar{\psi}$ if $\lambda_c \to 0,\ c \in \{c,\dots,C\}$.

4. $\forall\alpha > 0$, the critical value of every ψ is assumed to be finite and strictly smaller than the supremum value: $T'_\alpha < \bar{\psi}$.

b note that, in this context, partial tests are permutationally equivalent to their p-values: $T_c \overset{\pi}{=} \Pr\left\{T_c^* \geqslant T_c^{obs}|\mathbf{X}\right\} = \lambda_c,\ \ c = 1,\dots,C$

Note that these properties define a class Ψ of combining functions that contains for example:

- Fisher's: $T' = -2\sum_{c=1}^{C} \log p_c$, $0 \leqslant T' \leqslant +\infty$;
- Tippett's: $T' = \min_c p_C$, $0 \leqslant T' \leqslant 1$;
- Liptak's: $T' = \sum_{c=1}^{C} \Phi^{-1}(1 - p_c)$, where Φ is the standard normal cumulative distribution function, $0 \leqslant T' \leqslant +\infty$.

After obtaining the statistics of the second-order test T' the global p-value will be simply obtain with:

$$\lambda^G = \frac{1}{B}\sum_{b=1}^{B} \mathbb{I}\left\{T_b'^{*} \geqslant T'\right\}. \tag{2.6.2}$$

We refer to Pesarin and Salmaso (2010b) for further details about theory and applications of perumation tests and Nonparametric Combination Methodology.

2.6.4 Finite Sample Consistency

Another important feature of the NPC Methodology is the Finite-Sample Consistency (FSC), a property that can be described in the following simple manner. Under certain conditions, if the sample size is kept fixed and the number of variables increases, we have that the rejection rate (conditional and unconditional) of the combined test T converges to 1.

This means that if the null hypothesis is not true and we keep on adding informative variables to the dataset, the NPC-combined test is guaranteed to reject the null from a certain point onwards.

This characteristic is really important for our purposes as well as for applications, because often it is not possible to gather more samples in order to increase n but it may be feasible (although expensive) to measure more variables on the same set of samples.

The main conditions under which FSC is valid are the following, explained from the one-sided two-sample test point of view for simplicity.

(I) T is any associative test statistic for one-sided hypotheses;

(II) sample sizes (n_1, n_2) are fixed and finite;

(III) the observed data set is $\mathbf{X}(\delta) = \mathbf{Z}_1 + \delta, \mathbf{Z}_2$, where $(\mathbf{Z}_1, \mathbf{Z}_2) \in \mathcal{X}^n$ are i.i.d real random deviates with parent distribution P_Z and $\delta = (\delta_1, \ldots, \delta_{n_1})^\top$ is the vector of nonnegative fixed effects;

(IV) δ diverges according to a monotonic sequence $\{\delta_\nu, \nu \geqslant 1\}$, which elements are such that $\delta_\nu \leqslant \delta_\omega$ with $\nu < \omega$.

Note that all these conditions are fullfilled for the NPC-based test statistics considered in this work.

2.7 ITERATED NPC

The NPC relies on the chosen combining function and although it is independent from the underlying distribution, each combining function leads to different rejection regions and hence slightly different overall p-value. This does not pose a problem from the asymptotic point of view, due to their consistency under the alternative hypothesis (Pesarin and Salmaso, 2010b), but may be still useful to try to reduce the influence of the finite-sample behaviour of the specific combining function.

Indeed, each combining function has its own properties and is more suitable in some situations rather than others. For example Tippett's solution is desirable (in terms of good power behaviour) if we think that only one or few, but not all, sub-alternatives are true; Liptak's one is better if we expect that possibly all sub-alternatives are true; Fisher's behaviour instead lies between these two hence is desirable when no clear structure of the sub-alternatives is expected.

In order to "average-out" the effect of the specific combining function adopted we will make use of an iterated strategy that will be henceforth referred to as: *Iterated Nonparametric Combination*. The steps of this iterated approach are as follows.

1. Choose which combining functions are to be used in the process, in the following we will consider the case of 3 functions: *e.g.* Fisher's, Liptak's (normal CDF), and Tippett's (min p-value).[c]

2. Combine the partial tests with each of the combining functions, obtaining three vectors of test statistic suitable for the global test.

3. Transform the three test statistics vectors in permutation p-values (separately from each other), obtaining the p-values distribution of the global test coming from the three functions.

4. Combine these three vectors as if they were partial tests, again with each one of the chosen combining function.

5. Repeat steps 3 and 4 until convergence of the observed p-values.

Of course we will need a stop criteria that measures the convergence of the observed p-values, we can check either if all observed p-values are "close enough" to the ones in the previous step or just check if they are close to each other at the current step. In any case, from preliminary simulations it seems that convergence is reached really quickly (2 to 4 steps). Another aspect important to remember is that permutation p-values depend on the cardinality of the permutation space, hence even in the case of successful convergence, p-values are allowed to vary between each other of maximum $1/C$ where $C = \left|\mathcal{X}^n_{/\mathbf{x}}\right|$, *i.e.* the cardinality of the permutation space.

c Note that, although in this way we greatly reduced the dependence of the results from the specific combining function, the set of chosen combining function still influences the final result. Thus different sets can still lead to different results.

2.7.1 Stopping Rule

There are currently 4 kinds of stopping rule implemented in the package, the algorithm stops when the condition does not hold anymore; here $\varepsilon = \delta/C$, where δ is the desired tolerance and ε is the minimum attainable one.

"ABS": based on the absolute differences between two subsequent iterations, hereafter named respectively p'_i and p''_i (current iteration), with expression $\max_i |p'_i - p''_i| > \varepsilon, i = 1, 2, 3$;

"EDF": based on the empirical distribution function of the p-values, $|\bar{p}' - \bar{p}''| > 2\sqrt{\varepsilon \bar{p}'(1 - \bar{p}')}$, it is similar to a t-test;

"NORM2": simply the euclidean distance between two contiguous iterations, hence $\|p'_i - p''_i\|_2$;

"SSQ": the Sum of Squared differences of the p-values with their mean in the current iteration, $\sqrt{\sum_{i=1}^{3}(p''_i - \bar{p}'')^2} > \varepsilon$

3 CORE ALGORITHM

As mentioned in the introduction, the problem of stochastic ordering arises from the need of a realistic formulation allowing to compare $G \geqslant 2$ given populations (possibly multivariate) with the goal of ranking them.

In this situation the goal we wish to achieve, in terms of inferential analysis, is not only to determine whether the populations are equivalent against the alternative that they are different, but also which hypotheses are rejected and which are not because we need to quantify the relative preference of each population *w.r.t.* the others. Using these measures we would finally determine an order among these populations (groups or treatments in practical terms) and rank them from the "best" to the "worst".

3.1 FORMALIZATION OF THE PROBLEM

In order to introduce the main concepts, let us start from a simple MANOVA model with a minimum set of assumptions as in 2.6.1 on page 19.

Let then $\mathbf{Y}$ be the multivariate numeric variable related to the D-variate response of any experiment of interest and let us assume, without loss of generality, that high values of each marginal univariate component correspond to better performance and therefore to a higher degree of preference.

The experimental design of interest is defined by the comparison of G groups or treatments with respect to D different variables where n replications of a single experiment are performed by a random assignment of statistical units to treatments (or groups).

The G-group multivariate statistical model (with fixed or random effects) can then be represented as follows:

$$\begin{aligned} &\mathbf{Y}_{ik} = \boldsymbol{\mu}_k + \boldsymbol{\varepsilon}_{ik}, \boldsymbol{\varepsilon}_{ik} \sim \text{IID}(0, \boldsymbol{\Sigma}) \\ &\quad i = 1, ..., n; \quad k = 1, \ldots, G. \end{aligned} \tag{3.1.1}$$

Where here n represents the number of replications; index k is related to treatments with D-variate mean effect equal to $\boldsymbol{\mu}_k$; and $\boldsymbol{\varepsilon}_{ik}$ is a D-variate random term of experimental errors with zero mean and covariance matrix $\boldsymbol{\Sigma}$. Here, for the sake of simplicity, we consider the balanced design in which all groups have the same number of samples, therefore n, otherwise we should have used $n_k, k = 1, \ldots, G$, but the method is clearly applicable also in case of unbalanced design. The only assumption needed here is that $\boldsymbol{\mu}_k < \infty \ \forall\, k$.

Since the focus of this work is on stochastic ordering and pairwise comparisons we first need to define what is meant with these terms, we will follow the simpler structures in Shaked and Shanthikumar (2007).

In particular, to define the *multivariate stochastic order* we will start from the usual definition of univariate stochastic order.

Definition 3.1.1 (Univariate Stochastic Order)**.** *Following the usual stochastic order definition, we would say that group* k *is* stochastically smaller *than and* not equal *to group* h *in the* j*-th component variable if*

$$\begin{aligned} &F_{Y_{jk}}(x) = \Pr\left\{Y_{jk} \leqslant x\right\} > \Pr\left\{Y_{jh} \leqslant x\right\} = F_{Y_{jh}}(x) \\ &\qquad \forall x \in U, \quad \textit{and} \quad Y_{jk} \overset{d}{\neq} Y_{jh} \end{aligned} \tag{3.1.2}$$

where $U \in \mathfrak{R}$ *is a non-empty measurable set for which the strict inequality holds, and* $A \overset{d}{\neq} B$ *is the usual inequality in distribution,* i.e. $F_A \neq F_B$*, for* $n \to \infty$*. Note that* $\mathfrak{R}$ *can be substituted with* $\mathfrak{N}$ *or* $\mathfrak{Z}$ *in case of ordinal variable. This relationship will be indicated as* $Y_{jk} \overset{st}{\lneq} Y_{jh}$.

We will assess this by means of the corresponding set of hypothesis:

$$H^{j}_{0(k,h)} : Y_{jk} \overset{d}{=} Y_{jh} \quad \textit{vs.} \quad H^{j}_{1(k,h)} : Y_{jk} \overset{st}{\lesssim} Y_{jh}, \tag{3.1.3}$$

which will be referred to as *univariate pairwise hypotheses*.

The concept of *Multivariate Stochastic Order* that will be used, instead, is the most simple generalisation of the univariate one: as soon as at least one component variable is considered stochastically larger (smaller), meaning the rejection of the correspondent null hypothesis, the D-dimensional variable will be considered stochastically larger (smaller) as well. More formally:

Definition 3.1.2 (Multivariate Stochastic Order)**.** *The* D*-dimensional random variable* $\mathbf{Y}_h$ *is said to be* stochastically larger *than and* not equal *to* $\mathbf{Y}_k$ *if, in the following setting, the null hypothesis is rejected in favour of the alternative:*

$$\begin{aligned} H^{G}_{0(k,h)} &: \mathbf{Y}_k \overset{d}{=} \mathbf{Y}_h \Leftrightarrow \bigcap_{j=1}^{D} \left\{ Y_{jk} \overset{d}{=} Y_{jh} \right\} \\ H^{G}_{1(k,h)} &: \mathbf{Y}_k \overset{st}{\lesssim} \mathbf{Y}_h \Leftrightarrow \bigcup_{j=1}^{D} \left\{ F_{jk} > F_{jh} \right\}, \end{aligned} \tag{3.1.4}$$

where the equivalence between the two sides of the double-implication symbol, under certain conditions, has been proven in Basso et al. (2009), which we refer to for further details.

The former definition defines the concept of multivariate stochastic order by means of the univariate (marginal) ones but for our purposes another equivalent definition is more suitable becuase it involves a generic map performing a dimensionality reduction.

Definition 3.1.3 (ψ-Stochastic Order)**.** *The* D*-dimensional random variable* $\mathbf{Y}_h$ *is said to be* stochastically larger *than* $\mathbf{Y}_k$ *if we have that:*

$$\mathbb{E}\left[\psi(\mathbf{Y}_k)\right] < \mathbb{E}\left[\psi(\mathbf{Y}_h)\right], \quad \textit{for any} \;\; \psi : \mathcal{R}^D \mapsto \mathcal{R} \tag{3.1.5}$$

provided that all expectations exist and $\psi(\cdot)$ *is non-decreasing in any of its arguments.*

Note that the choice of the functions $\psi(\cdot)$ in the previous definition is particularly sensitive as it represents the way in which the data dimensionality is reduced. Yet if we employ as $\psi(\cdot)$ any combining function equipped with the properties defined in the previous chapter, it is easy to see that the definition holds. In particular, given that we will make use of directional test statistics that are significant for large values (*i.e.* assume large values under the alternative), there will be a direct (although not necessarily linear) relationship between the univariate test statistics, arguments of the combining function, and the expected value of the combination.

Of course it is often unfeasible and undesirable to prove 3.1.3 for *all increasing functions*, hence we will rely on the first definition 3.1.2 as soon as the thesis of the second is verified.

Note also that the two definitions do not exclude the situation in which some component variables are under the alternative in one direction and at the same time some other variables are active in the other direction. This is not undesirable, as we will see in the following sections, because the other direction will be dealt with as well before arriving to the final ordering.

3.2 THE TEST STATISTIC

The core element and smallest building block of the method that will be described hereafter is the test statistic for the univariate pairwise comparison, the choice of which can be related to the problem at hand and to the nature of the data being analysed.

In general the test statistic T related to the pairwise hypothesis (k, h) for the j-th variable will be of the form:

$$T_{j(k,h)} = T\left(\mathbf{y}_{jk}, \mathbf{y}_{jh}\right), \quad \text{with} \quad T : \mathcal{R}^{2n} \mapsto \mathcal{R}; \qquad j = 1, \dots d; \quad k \neq h; \quad k, h = 1, \dots G \tag{3.2.1}$$

where $\mathfrak{R}$ can be substituted with $\mathfrak{Z}$ or $\mathfrak{N}$ depending on the kind of data at hand, *i.e.* ordinal or continuous data. The main feature that this test statistic needs to have is to be sensitivive for large values against the null hypothesis.

Note that, as we refer to the model in Equation (3.1.1), test statistics presented henceforward are testing the relative location-shift rather than focussing on the whole distribution as would suggest the definitions given in the previous section. This is also the reason why in Chapter 4 we can define the true ranking by defining the group means.

In particular, at the moment the following tests statistics are coded in the `R` package that will be used for the MonteCarlo simulation, they are divided depending on the possible kind of data they can treat and according to the experimental design they are suited to analyse.

3.2.1 Test Statistics for Continuous Data

Hereafter are presented the test statistics that are already implemented in the `R` package `SOUP`, suitable for continuous data. We assume in the following that all variables that are taken into account are of the same type hence, in this case, they can all be considered continuous.

Modified Hotelling's T^2

This first example is a slightly modified version of the well-known Hotelling T^2 statistic for two sample comparisons, in which the covariance matrix of the quadratic form composing the statistic is the pooled one computed using all data, instead of just the two groups involved. We adjusted the statistic in order to compute the parametric p-values, *i.e.* we can use as reference distribution, under the null hypotesis, both the Fisher's $\mathcal{F}$ and the permutation one. Note that this kind of statistic can be used only with the simple experimental design in which there are no covariates apart from the one containing the labels of groups.

The expression of the test statistics is the following.

$$
\begin{aligned}
\widehat{T}^{H}_{(h,k)} &= c \cdot \frac{n^2}{2n} \left(\bar{\mathbf{y}}_h - \bar{\mathbf{y}}_k\right)^\top \cdot \hat{\mathbf{W}}^{-1} \cdot \left(\bar{\mathbf{y}}_h - \bar{\mathbf{y}}_k\right) \\
c &= \frac{nG - D - 1}{DG(n-1)}, \quad h \neq k, \quad h, k = 1, \ldots, G \\
\hat{\mathbf{W}} &= \frac{1}{G(n-1)} \sum_{i=1}^{nG} \left(\mathbf{y}_i - \bar{\mathbf{y}}\right) \cdot \left(\mathbf{y}_i - \bar{\mathbf{y}}\right)^\top \\
\widehat{T}^{H}_{hk} &\sim \mathcal{F}_{D,nG-D-1} \quad \text{under} \quad H^{G}_{0(k,h)}
\end{aligned}
\tag{3.2.2}
$$

Where c is the correction factor that relates $\widehat{T}^{H}_{hk}$ to the $\mathcal{F}$ distribution, and $\hat{\mathbf{W}}$ is simply the pooled covariance matrix under the global null hypothesis in which all data come from the same distribution.

Of course the test statistic just defined is unsigned so it is not suitable to test directional alternatives. To this aim we will make use of an additional step, matching each component variable to its direction. This step consists in the following substeps:

1. For each pairwise comparison, compute the following sign $s_{(h,k)} := \operatorname{sign}\left\{\left(\bar{\mathbf{y}}_h - \bar{\mathbf{y}}_k\right)^\top \hat{\mathbf{W}}^{-1/2} \mathbf{1}_D\right\}$, where $\hat{\mathbf{W}}^{-1/2}$ is the Cholesky decomposition of the inverse of $\hat{\mathbf{W}}$ and $\mathbf{1}_D$ is the D-dimensional vector of all ones.[a]

2. If asymptotic p-values are to be used, compute them *w.r.t.* the reference distribution, then complement to one any p-value with positive $s_{(h,k)}$. On the other hand, if permutation p-values are to be used, match the sign of each pairwise comparison $s_{(h,k)}$ with the corresponding test statistic.

In this way we obtain a signed version of the Hotelling's T^2 test statistic for pairwise comparison of multiple groups, the sign of which relies on a squared-root version of the same test statistic.

a Note that $\hat{\mathbf{W}}^{-1/2}$ is needed for computing the signs, as it weights variables according to their covariance matrix.

Means Difference And t-Test

Another test statistic that can be used with continuous data is simply the pairwise difference of the group means, hence in formula:

(3.2.3)
$$\widehat{T}^{md}_{j(k,h)} = \overline{y}_{ih} - \overline{y}_{ik} = \frac{1}{n}\sum_{i\in h} y_{ih} - \frac{1}{n}\sum_{i\in k} y_{ik}, \quad h,k = 1,\dots,G$$

where here the statistic is computed for each variable separately as it refers to the set of univariate hypotheses in Equation (3.1.3) on page 29, and this will be the case for most of the ones proposed in this work.

Or we can also use its scaled version, the two-sample t-test, where the $\hat{\sigma}^2$ in the denominator is the residual deviance coming from the fit of a one-way ANOVA model on that variable and it is assumed equal in the two samples:

(3.2.4)
$$\widehat{T}^{t}_{j(k,h)} = \frac{\overline{y}_{ih} - \overline{y}_{ik}}{\sqrt{\hat{\sigma}^2\left(\frac{1}{n}+\frac{1}{n}\right)}}, \quad h,k = 1,\dots,G$$

With the homoscedasticity and normality assumptions, the statistic follows a Student-t distribution with $G\cdot(n-1)$ degrees of freedom, under the null; this will be used to compute the parametric p-values.

Note that these two test statistics can be used also in presence of a categorical covariate with S categories, hence in case of a stratified design, although in this situation they have to be treated differently. Indeed, for the first one it is enough to keep the pairwise difference of means separated for each level of the categorical covariate (stratum), and then sum them together afterwards. For the second one instead, a two-way ANOVA has to be fitted in order to get the estimate of the residual variance $\hat{\sigma}^2$, and also the degrees of freedom for the Student-t reference distribution has to be changed to $G\cdot(n-1)-S$.

LINEAR INTERACTION A practical option, available for both these test statistic, can be useful for certain applications: if we expect the levels of the stratifying variable to be informative, like *e.g.* in case of number of cycles in a engineering stress-test, we can remove the linear effect of the covariate by dividing each observation by his startum level. The analysis is then carried out in the same way as before, as it safer to assume that the linear effect is not the only one present, yet this simple operation can benefit the analysis in case we have that information.

Linear Regression Based

This test statistic is specifically designed for the stratified analysis in which a categorical covariate is present and the experimenter expects a possibly nonlinear relationship between each one of the response variables and the stratifying covariate.

The need for this kind of test statistic comes from a specific class of industrial experiments in which secondary performances of washing machine detergents are tested. Primary performances measure the ability of the product in removing various kinds of soils, whereas secondary performances assess the ability of the detergent in *e.g.* "keeping the fabric white" or more generally "maintaining the colours of the fabric as they are" (colour fading, colour transfer), or again "avoiding calcium deposits on the heating resistances of the washing machine". For more details on motivations and applications we refer to the Master's Thesis of Gomiero (2010).

In order to compute the test statistic we need to start from the univariate linear model from which it is derived:

$$Y_{ijhs} = \beta_{0j} + \beta_{1jh} Z_{ih} + \sum_{q=2}^{Q} \beta_{qjs} X_{is}^{q-1} + \varepsilon_{ijhs}, \tag{3.2.5}$$
$$i = 1, \dots, n; \quad j = 1, \dots, d; \quad h = 2, \dots G; \quad s = 1, \dots, S.$$

Here Z_{ih} is the dummy variable coding for the group h with the usual convention that the first group acts as the baseline, X_{is} is the value of the stratifying covariate for observation i and

stratum s, with S being the total number of strata, and $Q-1$ is the degree of the polynomial with which we fit the effect of the covariate. It is an additive model for the group's effect and a polynomial regression on the stratifying covariate in which, for most of the practical cases, setting $Q = 3$ or 4 will be enough (quadratic or cubic polynomial). Thus we are trying to assess the effect of the groups while controlling for the covariate effects with a polynomial regression. Furthermore, note that also here we can modify the formula in order to introduce the linear interaction between the covariate values and the group's effect, leading to:

$$Y_{ijhs} = \beta_{0j} + \beta_{1jh} Z_{ih} \cdot X_{is} + \sum_{q=2}^{Q} \beta_{qjs} X_{is}^{q-1} + \varepsilon_{ijhs}, \tag{3.2.6}$$
$$i = 1, \dots, n; \quad j = 1, \dots, d; \quad h = 2, \dots G; \quad s = 1, \dots, S.$$

In this setting the univariate hypotheses are to be based on the β_{1jh} coefficients so the Equation (3.1.3) on page 29 needs to be rewritten as:

$$H^j_{0(1,h)} : \bigcap_{h=2}^{G} \{\beta_{1jh} = 0\} \quad \text{vs.} \quad H^j_{1(1,h)} : \bigcup_{h=2}^{G} \{\beta_{1jh} < 0\} \tag{3.2.7a}$$

for the comparisons with the baseline

(3.2.7b)

$$H^j_{0(k,h)} : \bigcap_{\substack{k<h \\ h,k\neq 1}} \{\beta_{1jk} = \beta_{1jh}\} \quad \text{vs.} \quad H^j_{1(k,h)} : \bigcup_{\substack{k<h \\ h,k\neq 1}} \{\beta_{1jk} < \beta_{1jh}\}$$

for the other comparisons

Here though, there is an additional complication due to the pairwise comparisons that do not involve the baseline group: while for the comparisons in Equation (3.2.7a) we can just use the usual t-test with $G(n-1)+Q-1$ degrees of freedom for the nullity of the parameters in the linear model, we need a constraint testing solution for the other comparisons in which

we substitute β with $R^\top\beta$, with $R^\top$ being the vector of constraints, basically a contrast vector for the two parameters to be compared. Leaving out the details (for which we refer again to Gomiero (2010)), the expression of the test suitable for a single pairwise comparison between β parameters is:

(3.2.8)
$$\frac{\hat{\varepsilon}_R^\top \hat{\varepsilon}_R - \hat{\varepsilon}^\top \hat{\varepsilon}}{\hat{\varepsilon}^\top \hat{\varepsilon}} \cdot k \sim \mathcal{F}_{1,k} \stackrel{d}{=} \mathbf{t}_k \quad \text{with} \quad k = G(n-1) + Q - 1$$

where $\hat{\varepsilon}$ is the residuals vector of the unconstrained model, and $\hat{\varepsilon}_R$ that of the constrained model.

Finally, the test statistic based on the parameters of this linear model has the expression:

(3.2.9) $$\widehat{T}^{lm}_{j(h,k)} = \widehat{\beta}_{1jh} - \widehat{\beta}_{1jk}, \quad h, k = 1, \ldots, G;$$

where the way p-values are computed depends on the comparison at hand, *i.e.* if it is a comparison with the baseline or not, but the reference distribution is the same thanks to the relationship between the Fisher's $\mathcal{F}$ and the Student's $\mathbf{t}$ distributions.

It is important to note that the tests in this section are not exact, as they are based on residuals from a (constrained) linear model.

3.2.2 Test Statistics for Categorical Data

Although using test statistics designed for continous data is possible with categorical data, and sometimes even a good approximation, it was worth to implement also another one, specifically designed for categorical data.

Directional Anderson-Darling

In order to deal with categorical data we implemented a modified version of the Anderson-Darling statistic (original papers: Anderson and Darling, 1952, 1954), which falls into the big fam-

ily of the "Goodness-of-Fit" tests based on the Empirical Distribution Function (EDF), and it was designed to test the proximity of a distribution to a reference one.

In our case we needed a directional version of the statistic suitable to compare the relative preference between two groups (see Pettit, 1976, for the two sample version, and Scholz and Stephens, 1987, for the K-sample extension), as it is the core element of the method described in this work. The fact that we rely on a permutation framework, though, makes it easier to define such test statistic because everything is *conditioned on the observed sample*.

In order to do this we have to start with the definition of the EDF for ordinal variables, that can be written as $\widehat{F}(q) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{I}\{y_i \leqslant q\}$ where q is the category and, without loss of generality, we can say it runs from 1 to Q. Now we can use the statistic to test how close the EDF of a single group is to the pooled EDF for all groups for each category.

In this way, though, we will not have a directional alternative becuase the original expression contains the squared difference between the two EDFs. In order to render it directional, we can simply use its square-root version so that we will have a signed-test for the equivalence of the single group EDF and the pooled EDF. If we now compute the difference between such test statistics, obtained from two different groups, *e.g.* h and k, we can consider it as a sort of EDF-based two-sample t-test and use it against the directional alternatives we are interested in.

In a more formal way, if we consider a single variable with Q categories and two groups being compared, the test statistic consists in the pairwise EDF difference of the two groups for each category, standardised by the variance of the pooled EDF. The expression of the statistic is therefore:

(3.2.10)

$$\begin{aligned}\widehat{T}^{AD}_{j(h,k)} &= \sum_{q=1}^{Q} w_{jq}\left(\widehat{F}_{jk}(q) - \widehat{F}_{j\bullet}(q)\right) - \sum_{q=1}^{Q} w_{jq}\left(\widehat{F}_{jk}(q) - \widehat{F}_{j\bullet}(q)\right) \\ &= \sum_{q=1}^{Q} w_{jq}\left(\widehat{F}_{jk}(q) - \widehat{F}_{jh}(q)\right) \\ \text{with} \quad w_{jq}^{-1} &= \sqrt{\widehat{F}_{j\bullet}(q)\left[1 - \widehat{F}_{j\bullet}(q)\right]}\end{aligned}$$

Where $n \cdot \widehat{F}_{jh}(q) = \sum_{i=1}^{n} \mathbb{I}\left\{y_{ijh} \leqslant q\right\}$ is the EDF for the group h in the variable j and $nG \cdot \widehat{F}_{j\bullet}(q) = \sum_{i=1}^{nG} \mathbb{I}\left\{y_{ij} \leqslant q\right\}$ is the pooled EDF for variable j. Note that also $\widehat{T}^{AD}_{j(h,k)}$ will be used to test the system of hypotheses in Equation (3.1.3) on page 29 as are all other test statistics.

3.2.3 Permutation Strategies

To obtain the permutation distribution of each pairwise univariate test statistic $\widehat{T}^{\bullet}_{j(h,k)}$ (where $\bullet$ can be t, md, lm or AD) we have to compute each test statistic not only on the observed dataset but also for each one of the permuted dataset. In order to maintain the dependence structure among the variables we must use the same permutation for each variable, therefore it is enough to permute *entire rows* of the data matrix.

More formally, let $\mathbf{Y} \in \mathcal{R}^{N \times d}$ be the original $N \times d$ data matrix where $N = n \cdot G \cdot S$, or $n \cdot G$ is the number of rows that depends on the presence of the stratifying covariate or not. Then, if $\widehat{T}^{\bullet}_{j(h,k)}$ is obtained from $\mathbf{y}_j = \left(y_{1,j}, y_{2,j}, \ldots, y_{N,j}\right)$ (column j of the matrix), the b-th permutation ${}^{b}\widehat{T}^{\bullet}_{j(h,k)}$ is obtained from the permuted data vector ${}^{b}\mathbf{y}_j = \left(y_{u_{1,b},j}, y_{u_{2,b},j}, \ldots, y_{u_{N,b},j}\right)$ with $\mathbf{u}_b = (u_{1,b}, \ldots, u_{N,b})^{\top}$ being the b-th permutation of row indices.

In particular, in case of the simpler experimental design, in which there is no covariate to take into account we will have

$$\mathbf{u}_b = (u_{1,b}, \ldots, u_{nG,b})^{\top} = \boldsymbol{\pi}\{1, \ldots, nG\} \tag{3.2.11}$$

(where $\pi\{\cdot\}$ is the permuting operator), *i.e.* simply a shuffle (sampling without replacement) of the sequence $\{1, \dots, nG\}$. In case a stratifying covariate is present instead, permutations will have to be performed in blocks, hence restricted to within each *stratum* and different/independent from one stratum to the other, $\mathbf{u}_b$ will then have the following structure.

$$
\begin{aligned}
&\mathbf{u}_b = \left({}^1\mathbf{u}_b^\top, \dots, {}^S\mathbf{u}_b^\top\right)^\top \quad \text{with} \\
&{}^s\mathbf{u}_b = ({}^su_{1,b}, \dots, {}^su_{nG,b})^\top = \boldsymbol{\pi}\{1, \dots, nG\} \quad \text{and} \\
&{}^s\mathbf{u}_b \neq {}^r\mathbf{u}_b \quad \forall s \neq r
\end{aligned} \tag{3.2.12}
$$

3.3 THE RANKING ALGORITHM

After the choice of the test statistic and having computed all univariate pairwise test statistics we need to combine them together in order to obtain the multivariate test statistic suitable to test the system of hypotheses in Equation (3.1.4) on page 29 ($H^G_{0(k,h)}$). This of course does not concern the modified Hotelling's test statistic as it is designed to give directly a test for the global pairwise hypothesis.

This is possible with the Nonparametric Combination, indeed after having chosen a combining function or after having performed the iterated combination procedure we will have:

$$
\widehat{T}^{\bullet}_{(h,k)} = \psi\left(\widehat{T}^{\bullet}_{1(h,k)}, \dots, \widehat{T}^{\bullet}_{j(h,k)}\right), \quad h \neq k, \quad h, k = 1, \dots, G \tag{3.3.1}
$$

and its corresponding permutation distribution

(3.3.2)

$$\begin{pmatrix} {}^{1}\widehat{T}^{\bullet}_{(h,k)} \\ \vdots \\ {}^{b}\widehat{T}^{\bullet}_{(h,k)} \\ \vdots \\ {}^{B}\widehat{T}^{\bullet}_{(h,k)} \end{pmatrix} = \begin{pmatrix} \psi\left({}^{1}\widehat{T}^{\bullet}_{1(h,k)}, \ldots, {}^{1}\widehat{T}^{\bullet}_{j(h,k)}\right) \\ \vdots \\ \psi\left({}^{b}\widehat{T}^{\bullet}_{1(h,k)}, \ldots, {}^{b}\widehat{T}^{\bullet}_{j(h,k)}\right) \\ \vdots \\ \psi\left({}^{B}\widehat{T}^{\bullet}_{1(h,k)}, \ldots, {}^{B}\widehat{T}^{\bullet}_{j(h,k)}\right) \end{pmatrix}, \quad \begin{matrix} h \neq k, \\ h,k = 1,\ldots,G \end{matrix}$$

where here again $\bullet = md, t, lm$, or AD.

The next step is just the computation of permutation p-values for each one of these test statistics:

(3.3.3)

$$\lambda_{(h,k)} = \frac{1}{B}\#\left\{{}^{b}\widehat{T}^{\bullet}_{(h,k)} \geqslant \widehat{T}^{\bullet}_{(h,k)}\right\} = \frac{1}{B}\sum_{b=1}^{B}\mathbb{I}\left\{{}^{b}\widehat{T}^{\bullet}_{(h,k)} \geqslant \widehat{T}^{\bullet}_{(h,k)}\right\}$$
$$h \neq k, \quad h,k = 1,\ldots,G$$

where, for simplicity, we dropped the notation about the type of test statistic it originates from.

3.3.1 Multiplicity Adjustment

At this point we have gathered the information from the D variables and obtained a test for the relative preference of each group in comparison with all the others, but we still need a rule that allows us to exploit this information to construct the final ranking. In view of this it is useful to note that the p-values we have just obtained needs to be corrected for multiplicity because, even just looking at the logical dependencies, we are performing $G \cdot (G-1)$ tests based on G groups, hence we can not consider them independent. Indeed, even with $G = 3$ groups we can see that the resulting tests $(1,2)$, $(1,3)$ and $(2,3)$ will necessarily present some kind of dependence.

That being said, we can split in two the p-values multiplicity adjustment given that the tests for the comparisons of the type (h,k) are all testing the violation of the null hypothesis in one direction, whereas the ones for the comparisons of the type

(k, h) are testing against the other direction. Thus we just have to choose a multiplicity correction and perform it separately on the $G \cdot (G-1)/2$ tests against one direction, and on the $G \cdot (G-1)/2$ tests against the other direction.

Of course the kind of multiplicity correction depends on the desires of the experimenter and the needs of the specific study, as it can influences the results when *e.g.* differences between groups are not dramatic.

Popular quantities that the experimeter might want to control are *e.g.* FamilyWise Error Rate (FWER) and False Discovery Rate (FDR), where here "Family" refers to the collection of pairwise hypotheses against one direction; their definition is given in the following.

FWER FamilyWise Error Rate is defined as $\Pr\{V > 0 | H_0\}$ where V is the number of true hypotheses wrongly rejected;

FDR False Discovery Rate is defined as $\mathbb{E}\{V/R | R > 0\} \cdot \Pr\{R > 0\}$, hence the expected proportion of false rejections in the set of all rejections.

Two algorithms for the control of the FWER (the more strict one) are implemented in the `SOUP` package and are referred to as: "Bonferroni-Holm-Shaffer", or "BHS" (Shaffer, 1986), and "FWE-minP" (Westfall and Young, 1993, reprised within the context of NPC in Finos et al., 2003 and Pesarin and Salmaso, 2010b, on page 272).

The first one is a modification of the sequential rejection strategy of Bonferroni-Holm in which at each step i instead of multiplying our p-values $p_{(i)}$ by $(n - i + 1)$, where $p_{(1)} \leqslant p_{(2)} \leqslant \cdots \leqslant p_{(K)}$ and K is the number of hypotheses to be tested, we multiply the p-values by the maximum number of possibly true hypotheses (not rejected) at step i, given that at least $i - 1$ hypotheses are false (rejected) and taking into account the logical dependencies. The algorithm can be described as follows.

- Let $p_{(1)} \leqslant p_{(2)} \leqslant \cdots \leqslant p_{(6)}$ be the ordered p-values suitable for the $K = G \cdot (G-1) = 6$ pairwise tests coming from $G = 4$ groups and let α be the desired significance level.

- If we have that $p_{(1)} \cdot 6 > \alpha$ (step $i = 1$) then no hypothesis is rejected as the p-values are increasingly ordered.
- If we have instead that $p_{(1)} \cdot 6 \leqslant \alpha$ then *at least other two hypotheses must be rejected*, since if any two groups differ, at least one of these must differ from the remaining ones.
- Hence, in this settings, step $i = 2$ will be $p_{(2)} \cdot 3 \leqslant \alpha$ instead of $p_{(2)} \cdot 5 \leqslant \alpha$ of the Bonferroni-Holm procedure, and so forth.
- These sets of "maximum number of possibly true hypotheses" (the number 3 in the example), are actually the cardinality of subsets (partitions) induced by the rejection of some of the hypotheses in a pairwise comparison setting. They have to be computed recursively with the following formula: $C_0 = C_1 = \{0\}$, and $C_k = \bigcup_{j=1}^{k} \left\{ \binom{j}{2} + x : x \in C_{k-j} \right\}$ for $i \geqslant 2$.

The second one also stems from the Bonferroni-Holm procedure but relies on resampling considerations and more specifically on the permutation null distribution of the minimum p-value. Also this procedure ensure the strong control of the FWER and it can be described as follows:

1. let $p_{(1)} \leqslant p_{(2)} \leqslant \cdots \leqslant p_{(K)}$ be the vector of increasingly ordered p-values;
2. let $p'_{(i)} := \Pr\left\{\min_{j\in\{i,\ldots,K\}}(p_j)^* \leqslant p_{(i)}\right\}$ where $\min_{j\in\{i,\ldots,K\}}(p_j)^*$ denotes the permutation distribution of the minimum p-value and $p_{(i)}$ the observed minimum p-value;
3. if $p'_{(1)} \leqslant \alpha$ reject the corresponding hypothesis and go on, otherwise retain hypotheses from (1) to (K) and stop;
4. for $i = 2, \ldots, K$, $p'_{(i)} = \max\left(p'_{(i)}, p'_{(i-1)}\right)$ and if $p'_{(i)} \leqslant \alpha$ reject also the hyphothesis (i) and continue, otherwise retain hypotheses from (i) to (K) and stop.

As for controlling the FDR, two popular methods are already implemented in R : Benjamini and Hochberg (1995), and Ben-

jamini and Yekutieli (2001); but in this case the first two methods seemed more appropriate, especially because of the permutation framework and because we are dealing with pairwise comparisons leading to logical implications *w.r.t.* rejections.

3.3.2 Ranking Rule

Now our set of adjusted p-values $\lambda^{adj}_{(h,k)}$, $h \neq k$, $h, k = 1, \dots G$ can be used to build a $G \times G$ matrix that will be our starting point for the ranking rule, with the convention that $\lambda^{adj}_{(h,h)} = 1$. The matrix has the following structure:

$$\mathbf{\Lambda} = \begin{bmatrix} 1 & \lambda^{adj}_{(1,2)} & \cdots & \lambda^{adj}_{(1,G)} \\ \lambda^{adj}_{(2,1)} & 1 & \cdots & \lambda^{adj}_{(2,G)} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda^{adj}_{(G-1,1)} & \cdots & 1 & \lambda^{adj}_{(G-1,G)} \\ \lambda^{adj}_{(G,1)} & \cdots & \lambda^{adj}_{(G,G-1)} & 1 \end{bmatrix}. \tag{3.3.4}$$

Note that $\lambda^{adj}_{(k,h)} + \lambda^{adj}_{(k,h)} = 1$ is not necessarily true nor it is a problem because we are using these quantities as a kind of summarised score rather than formal p-values.

Now, by recalling that $\lambda^{adj}_{(k,h)}$ is associated with the global pairwise hypothesis (see Equation (3.1.4) on page 29), with alternative $\mathbf{Y}_k \overset{st}{\lesssim} \mathbf{Y}_h$, we can interpret each entry of the matrix, in football terminology, as the *score* resulting from a *match between group* h *and group* k.

Hence, continuing the parallel with the football world,

REJECTING the pairwise hypothesis $H^G_{0(k,h)}$ can be interpreted as "the group k has lost in a match against the group k"; whereas

NOT REJECTING $H^G_{0(k,h)}$ can be interpreted as "the match between group k and h was a draw".

From this point of view, looking at a specific row or column aquires an interesting meaning: if we focus on *e.g. row* k we can interpret it as containing all matches performed by group

k, and therefore all matches for which $H^G_{0(k,\bullet)}$ is rejected are the ones *lost* by that group; on the contrary, if we focus on *column* h we have the opposite interpretation, hence all matches *won* by group h are the ones for which $H^G_{0(\bullet,h)}$ is rejected.

If we now choose a desired significance level, and we keep the abovementioned interpretation in mind, we can build the final ranking in the simple following manner:

- let $L_k := \sum_{h=1}^{G} \mathbb{I}\left\{2 \cdot \lambda^{adj}_{(k,h)} \leqslant \alpha\right\}$, be the number of matches lost by group k, $k = 1, \dots G$;
- let $W_h := \sum_{k=1}^{G} \mathbb{I}\left\{2 \cdot \lambda^{adj}_{(k,h)} \leqslant \alpha\right\}$, be the number of matches won by group j, $h = 1, \dots G$;
- then the final ranking R_i of group i is simply $R_i = \mathbb{R}(W_i - L_i)$ where $\mathbb{R}(\cdot)$ is the usual rank operator, with the convention that the "best" (stochastically highest) group gets $R_i = 1$.

REMARK: note that the adjusted p-value $\lambda^{adj}_{(k,h)}$ is multiplied by 2 as we are now using it to test both directional alternatives simultaneously. Indeed all values *under* the main diagonal of the matrix are testing against the direction $\leqslant$ whereas values *over* the main diagonal are testing against the direction $\geqslant$; thus, if we consider a specific row or column, we are basically performing a two-sided test.

A Remark on Logic Transitivity

Let $G = 3$ the number of groups being tested, α be the desired significance level, and $(0,1]^{3\times 3} \ni \mathbf{\Lambda}$ be the resulting pairwise global test matrix. If we consider *e.g.* the first row, in practical cases we may have the following situation:

$$\left\{2 \cdot \lambda^{adj}_{(1,2)} > \alpha\right\} \wedge \left\{2 \cdot \lambda^{adj}_{(1,3)} \leqslant \alpha\right\} \wedge \left\{2 \cdot \lambda^{adj}_{(2,3)} > \alpha\right\}.$$

In this case there seems to be an inconsistency because, if we were to interpret the λs as p-values, we would say that: "group 1 is equivalent to group 2", "group 1 is stochastically smaller than group 3", and "group 2 is equivalent to group 3"; leading

to an apparent violation of the logic transitivity property of the order relation. In actuality λs are not seen as p-values but are regarded as *scores* where $\alpha/2$ acts as a threshold to determine if the match has to be considered a draw or not. Thus, avoiding the logical inconsistency, our interpretation of the previous situation would simply be that "the match between group 1 and 2 was a draw", "group 1 lost against group 3", and "the match between group 2 and 3 was a draw".

4 SIMULATION STUDY

In this chapter we will describe the simulation study that has been performed in order to empirically validate the proposed method, as well as assess the practical and computational characteristics of the algorithm and its applicability as a routine operation.

4.1 SIMULATION SETTINGS

With these simulations we want to study the behaviour of the procedure, both under the null and under some different structures of alternative hypothesis that seemed interesting from the statistical point of view. Moreover, we wanted it to be quite extensive so we have let vary not only the sample size or the number of variables, but also the number of groups, the distribution of the error term, and the nature of the data values (continuous or ordered categorical).

It is important to note that, in this conditional context, special attention is needed when defining the error term, because an improper definition of it may render the method unable to distinguish the groups.

For example, let us consider the simplest case of standard normal error with: $D = 1$ (univariate), group "h" under the alternative with $\mu_h > 0$, and group "k" under the null hypothesis (*i.e.* $\mu_k = 0$). In this case a μ_h equal to *e.g.* 1 might be too small to be considered relevant by the method as $\Pr\{Y_h < Y_k\} = \Phi_{1,2}(0) \cong 0.24$, where $\Phi_{1,2}(\cdot)$ is the distribution function of $\mathcal{N}(1, 2)$; *i.e.* the unconditional probability of rejection (power), may be not enough for the permutation test to attain a reasonably significant permutation p-value, and this applies with

different degrees of importance to all kind of errors and test statistics.

Getting back to the simulation settings and referring to the model in Equation (3.1.1) on page 28, we have the following general parameters:

MC number of MonteCarlo replications of each setting is 500;

G (number of groups) varies in $\{4, 6, 8\}$;

S (number of strata) will be equal to 1 for the non-stratified case and will be fixed to $S = 5$ for the stratified design.

n (number of replications) taking values $\{5, 11, 31\}$, hence from really low number of replications to quite large (for permutation tests);

D (number of variables) varies in $\{7, 13, 37\}$.

REMARK Note that not *all combinations* of settings will be tested as they are not all equally informative, especially with an eye on the applications, and also because the cartesian product of all sets of parameters would be a considerably big number. Indeed, identifying for a moment the set with the corresponding symbol (details are presented in the next subsection), $|G| \times |S| \times |n| \times |D| \times |\varepsilon| \times |\mu_k| = 3 \times 2 \times 4 \times 4 \times 4 \times 4 = 1536$ combinations of parameters in total.

Subsequently, we divide the simulations in two parts, according to the kind of data that are generated.

4.1.1 Continuous Data

Given that most of the test statistics described in this work are designed for continuous data, the majority of simulations will be on this type of data. In particular, the distribution of the error terms ε will be important and this is why four different ones will be considered:

1. $\mathcal{N}(0, 1)$, it acts as the benchmark distribution;

2. $\mathcal{N}(0, \Sigma)$ where $\{\Sigma\}_{i,j} = 0.7^{|i-j|}$, it basically introduces a long-memory serial dependency among variables, it is considered in order to have a slightly correlated set of variables;

3. $\mathbf{t}_2$, a heavy-tailed distribution;

4. $\Gamma(3/2, 1)$, highly asymmetric and positive distribution.

There are instead four scenarios for the vectors of group mean $\mathcal{R}^D \ni \mu_k, k = 1, \ldots, G$ each one designed to test one specific feature of the procedure.

1. All $\mu_k = 0\ \forall k$, therefore all the variables are under the global null hypothesis.

2. $\|\mu_2\|_2 = \frac{1}{2}\|\mu_{G-1}\|_2 > 0$ and $\|\mu_k\|_2 = 0$ for $k \neq \{2, G-1\}$; here we try to test whether the method can pick up correctly only two groups under the alternative while the rest of groups should get the same lower rank.

3. The same structure as the previous scenario but applied only on 3 variables, the rest are under the global null hypothesis; this is useful to check how much "noisy" variables (under the null) we can add before the method start to breakdown classifying all groups together.

4. $\|\mu_k\|_2$ linearly increasing with k; this scenario serves to test how well the method is able to give the correct ranking when all pairwise hypothesis are to be rejected in one direction or the other, *i.e.* when each group is different from all the others.

For all considered combinations of the settings, we will use as test stastistics: modified Hotelling's T^2, mean difference with permutation p-values, and the t-test like test statistic with asymptotic p-values.

4.1.2 Categorical Data

For the ordered categorical data case, as we have only one test statistic to be evaluated, we will consider all combinations of sample sizes and number of variables as well as two different types of errors. Moreover, each variable takes values in the set of integers between 1 and 10 to emulate the results of a questionnaire experiment or any other survey based on a set of answers with a fixed number of possible values. Note that all values are rounded to the next integer and shifted in order to take values between 1 and 10.

1. $\mathcal{N}(3,1)$, it acts as the benchmark distribution;

2. $10 \cdot \mathcal{B}(2,5)$, a beta distribution skewed towards the lower bound.

Of course the errors have been tuned such that they don't shift the values of the variables too much on the boundaries.

As for the alternative hypothesis, only one scenario will be tested: a modified version of the scenario in which means of groups are linearly increasing with k; the modification being the discretization of the group means before adding them to the pure errors data, and bounding the resulting values in the interval 1–10.

4.2 SIMULATIONS RESULTS

As the main interest lies in correctly ordering the groups under study, it is reasonable to check the final ranking for each setting combination across MonteCarlo replications. In particular, under the null hypothesis it makes sense to check only if all groups have received rank equal to 1 or not (global null hypothesis not rejected or rejected, respectively), whereas for the other scenarios it is more meaningful to check for each group if it got the correct rank or not. Hence the graphs that will be reported

in the following are divided by the scenario under which they fall.

The problem here is in the high number of graphs that should be drawn: one for each test statistic, for each sample size, and for each number of groups; that would be too much to convey the information efficiently and clearly. This is why we will make use of an additional level of "summarization" in between: one version of it for the null hypothesis and another one for the alternative hypothesis scenarios.

4.2.1 Under The Global Null Hypothesis

To explain this summarization we have to start from the graph of a single combination of simulation settings: figure 1 on the next page shows such a graph, where the proportion of rejections (across simulations) are plotted against the level of significance α. From this figure we compute the discretized version of the highlighted area and we standardize it with the total area under the bisector line; this value will be henceforth referred to as "Disagreement Area" or DA for simplicity.

This is the element we can use for our purposes as it contains the summarized information about the simulation: if the curve in the graph lies *all under* the bisector line, the procedure is *conservative* and the DA will be a positive value, whereas if the curve in the graph *goes over* the bisector line, it means the procedure is *liberal* (at least for some values of α) and the DA will have a negative value. Basically the DA measures the "distance" from the line of perfect agreement between theoretical and obtained coverage.

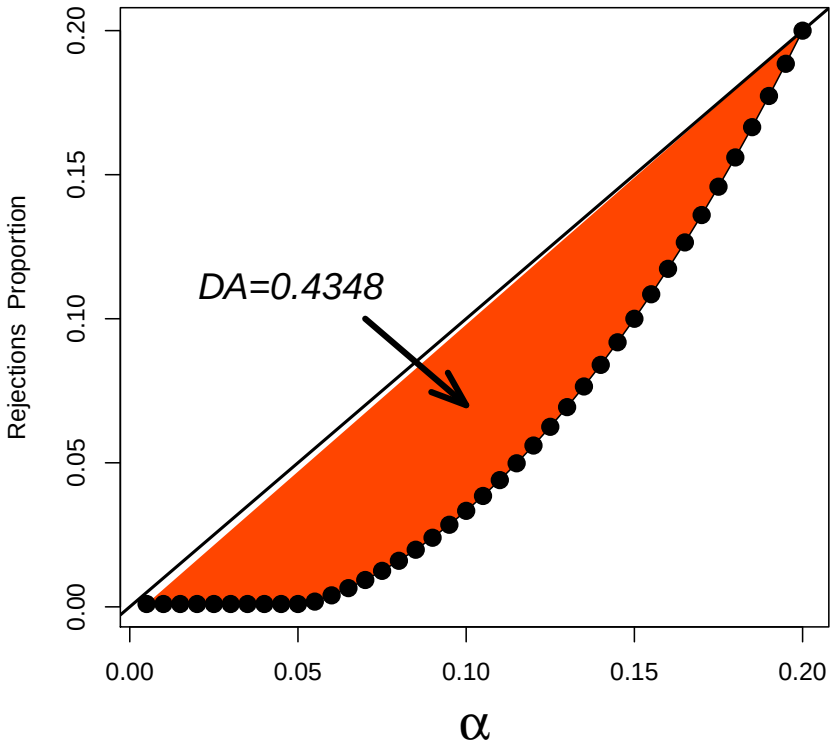

Figure 1: Toy example: results of a single simulation settings combination under H_0^G, it shows the proportion of rejected null hypothesis across MC simulations *vs.* the α significance level. The *Disagreement Area* (DA) is highlighted in red and has the interpretation "the lower in absolute value, the better".

The graphs that follow (Figures 4 to 5 on pages 56–57) contain the summary of all simulations under the null hypothesis where each dot is an DA value coming from one combination of simulations settings. More in detail, in the figures we have:

- values of DA *vs.* sample sizes n;
- 4 or 2 curves in each panel, *i.e.* one for each kind of error, for the continuous data and for the discrete data respectively;
- one panel for each number of groups and for each number of variables (9 panels in total);
- one graph for each test statistic.

Note that dots for sample size equal to 5 with 37 variables are absent in the Hotelling's test statistic for the reference distribution has no degrees of freedom in that situation.

Note also that lines marked by the symbol "1" are related to the standard normal errors where iterated combination had been used in place of usual single NPC.

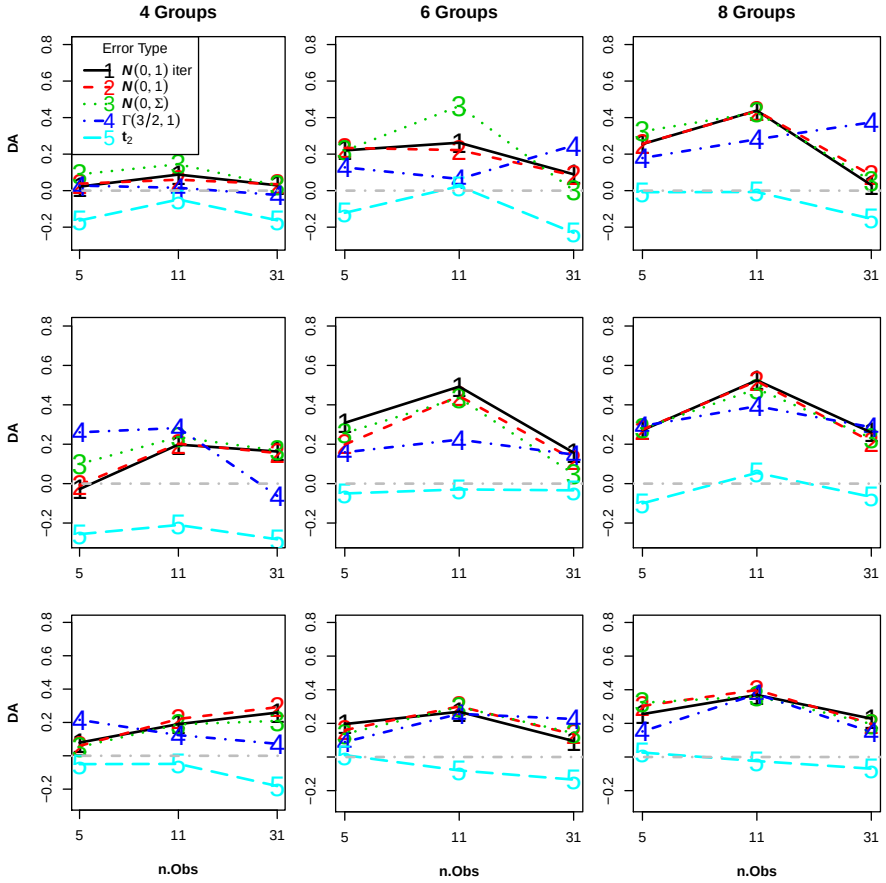

Figure 2: DA graph for the **"mean difference"** test statistic, each row is obtained with a different number of variables: 7, 13, and 37 respectively (top-down direction). It is slightly conservative in almost all situations, apart from when the error variance is large where it becomes slightly liberal.

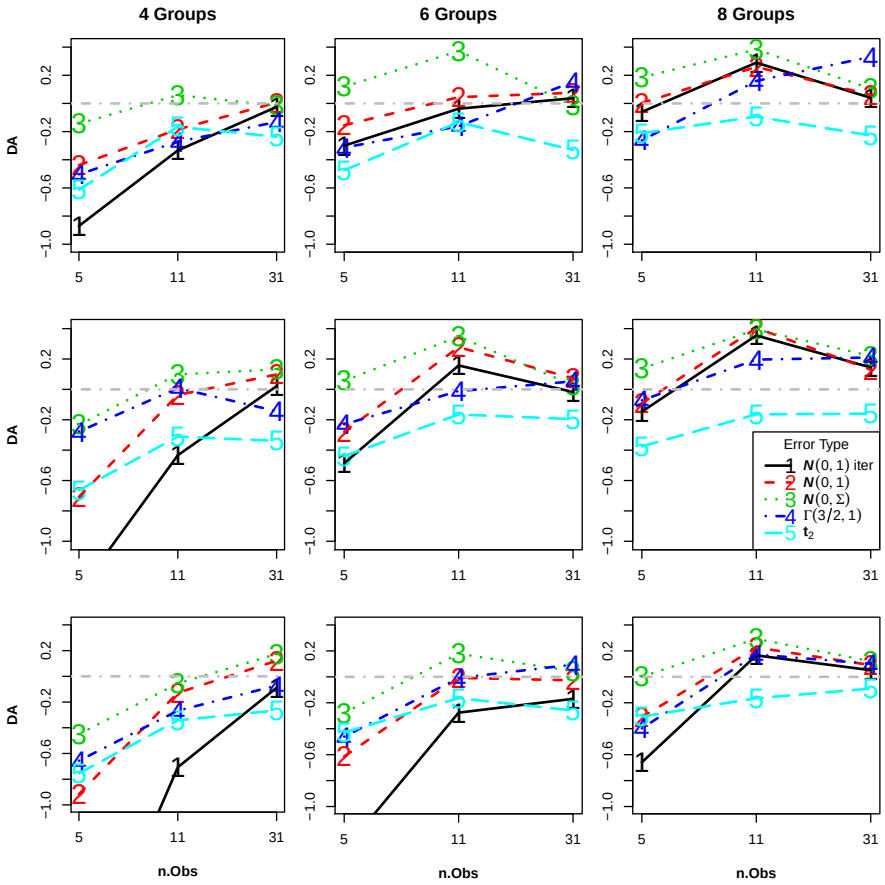

Figure 3: DA graph for the **"t-test"** test statistic, each row is obtained with a different number of variables: 7, 13, and 37 respectively (top-down direction). It is slightly liberal for small sample sizes and with 4 groups but has a quite good behaviour in all other situations.

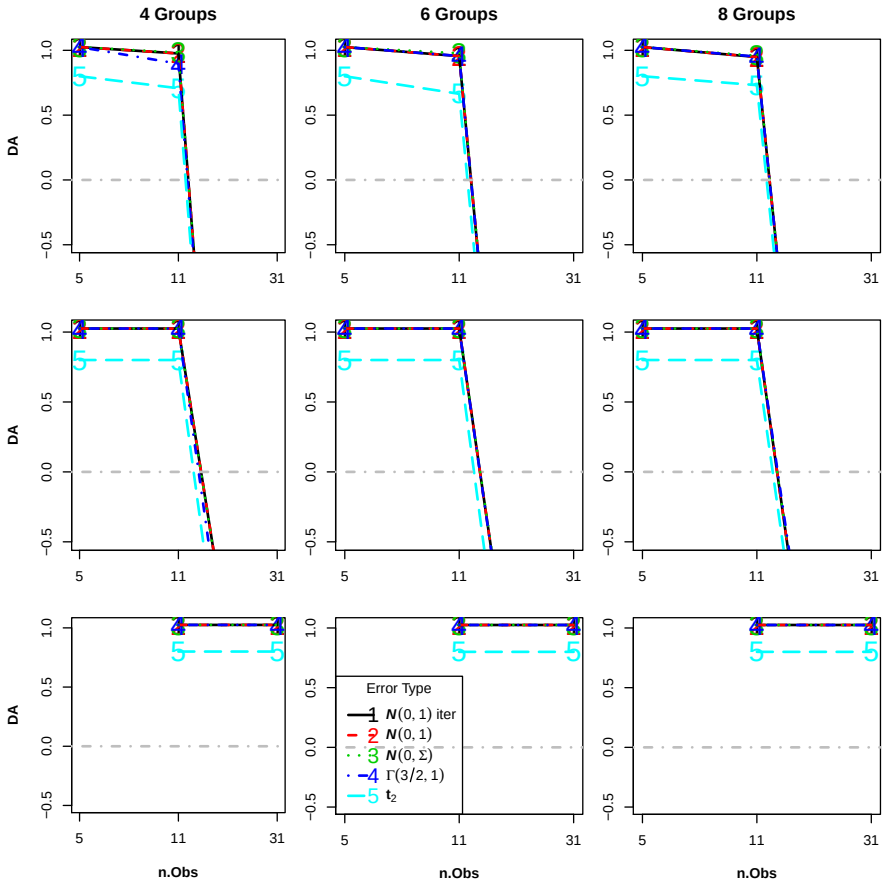

Figure 4: DA graph for the **"Hotelling's"** test statistic, each row is obtained with a different number of variables: 7, 13, and 37 respectively (top-down direction). The "breakdown" noticeable at 31 observations is due to an overfitting problem, this is why with 37 variables the results are basically the same as the other graphs.

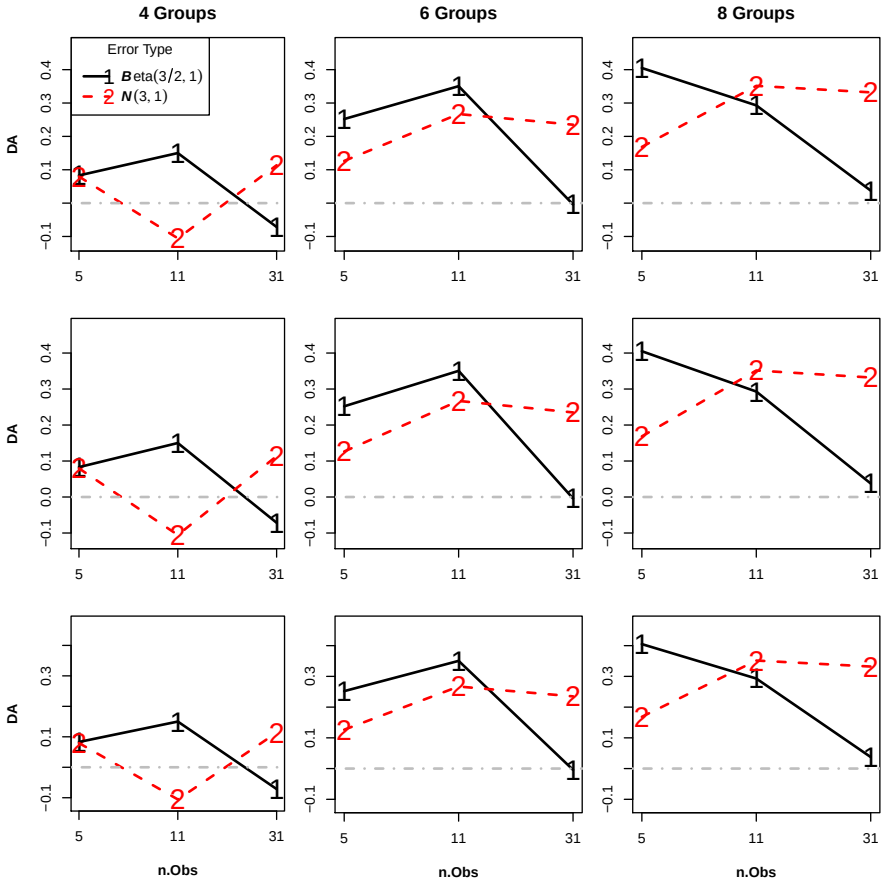

Figure 5: DA graph for the **"Anderson-Darling's"** test statistic, each row is obtained with a different number of variables: 7, 13, and 37 respectively (top-down direction).

Discussion

For *continuous data*, as we can see from the graphs, the test statistic that seems to have better coverage (*i.e.* better respect the α-level), is the "mean difference" one. Indeed, it is slightly conservative with almost all kinds of error, all numbers of groups, variables, and sample sizes; it becomes slightly liberal only when the error variance is large, hence with the $\mathbf{t}_2$ errors, but in general it behaves better than the other two.

The "t-test" test statistic is comparable to the "mean difference" one but lies more on the liberal side for most of the error kinds and especially for the smallest sample size $n = 5$.

Finally, the "Hotelling's" test statistic is the one with the worst behaviour because it is highly conservative with the two lower sample sizes (basically it rejects almost nothing), and becomes highly liberal with the largest sample size. The conservative behaviour is most likely due to the few degrees of freedom of this test statistic: it needs the sample size to be much greater than the dimensionality D to attain nominal levels. The strange liberal behaviour seen with $n = 31$, instead, is most likely related to numerical errors.

For *discrete data* instead, the "Anderson-Darling's" test statistic has a behaviour comparable to the "mean difference" one for continuous data: it is slightly conservative in almost all situations and when it crosses the zero line (becoming liberal) it does not take values lower than -0.1, that is, the area over the line of perfect agreement is roughly ten percent of the half-square area.

4.2.2 Under the Alternative Hypotheses

Under the alternative hypothesis scenarios the summarization step is more complicated as we have to check, in each simulation, if each group got the right rank.

A possible way to summarize this concept of *"closeness to the exact ordering"* is to compute the *distance* of the obtained ranking from the exact ranking. For example, we can use the $\mathcal{L}_1$

distance (or Manhattan distance) that has the following expression:

$$d_1(\mathbf{x}^0, \hat{\mathbf{x}}) := \|\mathbf{x}^0 - \hat{\mathbf{x}}\|_1 = \sum_{i=1}^{G} |x_i^0 - \hat{x}_i|, \tag{4.2.1}$$

where $\hat{\mathbf{x}}$ represents the estimated ranking vector and $\mathbf{x}^0$ the true ranking. The interpretation of this distance is clearly "the lower the better" and although it depends on the number of groups G, it is easy to compute the maximum distance we can obtain, as the worst case-scenario is when the method gives rank 1 to all groups. For example, for the fourth scenario in 4.1.1 on page 49, we have:

$$d_1(\mathbf{x}^0, \mathbf{x}^*) = \sum_{i=1}^{G} |1 - i| = \frac{(G-1)G}{2}, \tag{4.2.2}$$

this maximal value will be used to normalise the index such that all graphs will be on the same scale, and hence comparable.

Now that we have this measure, we can use it in our summary graphs which are reported in the following, separately for each test statistic.

More in detail, in order to assess both the *average closeness* to the true ranking and how many times the method estimated correctly the true ranking, we will have two series of graphs: one where the average of d_1 across simulations is plotted against the α significance level, the other one where the proportion of *attained exact* ranking is drawn *vs.* α.

In particular, we will have for each figure and for each alternative hypothesis scenario:

- either the average of d_1 measures across simulations or the proportion of times across simulations where the minimum d_1 was attained (*vs.* α);

- one curve for each error type in each panel, 2 for discrete data and 4 for continuous;

- given one specific row, 3 panels obtained with the same sample size, one panel for each number of groups G.

- one figure for each number of variables and for each test statistic (not all will be shown).

We have $2 \times 3 \times 3 \times 3 + 2 \times 3 =$ (Average and Exact ranking) $\times$ (# continuous data test statistics) $\times$ (# of variables) $\times$ (# of scenarios) $+$ (Average and Exact ranking for the AD test statistic for discrete data) $\times$ (# of variables) only for the 4th alternative scenario (see 4.1.1 on page 49). Thus, in total, we can produce $2 \times 3^3 + 6 = 60$ graphs although not each one of them is highly informative or different from the others.

Indeed, fortunately enough, lot of these graphs are really similar as they share lot of the settings, so we will be able to show only results obtained with 7 variables without losing important information.

Moreover, for the sake of clarity, we will show only graphs that convey some actual information in comparison with the others and omit the ones that can be regarded as identical, some of these will be put in the appendices for the interested reader to check.

Second and Fourth Scenarios

Here we group together results for the 2nd and the 4th scenarios as they produce really similar graphs for all situations and all test statistics. Recall that the fourth scenario is the one in which group averages are linearly increasing with the group index (*i.e.* $\mu_1 < \cdots < \mu_G$) for all variables; whereas the second scenario is the one where only μ_2 and μ_{G-1} are greater than 0 and the rest is under the null, also for all variables.

This can be probably explained by the fact that group means were shifted for all variables, so the amount of information was apparently high enough for the method to distinguish among them, whereas in the 3nd scenario the power goes down as only 3 variables are active.

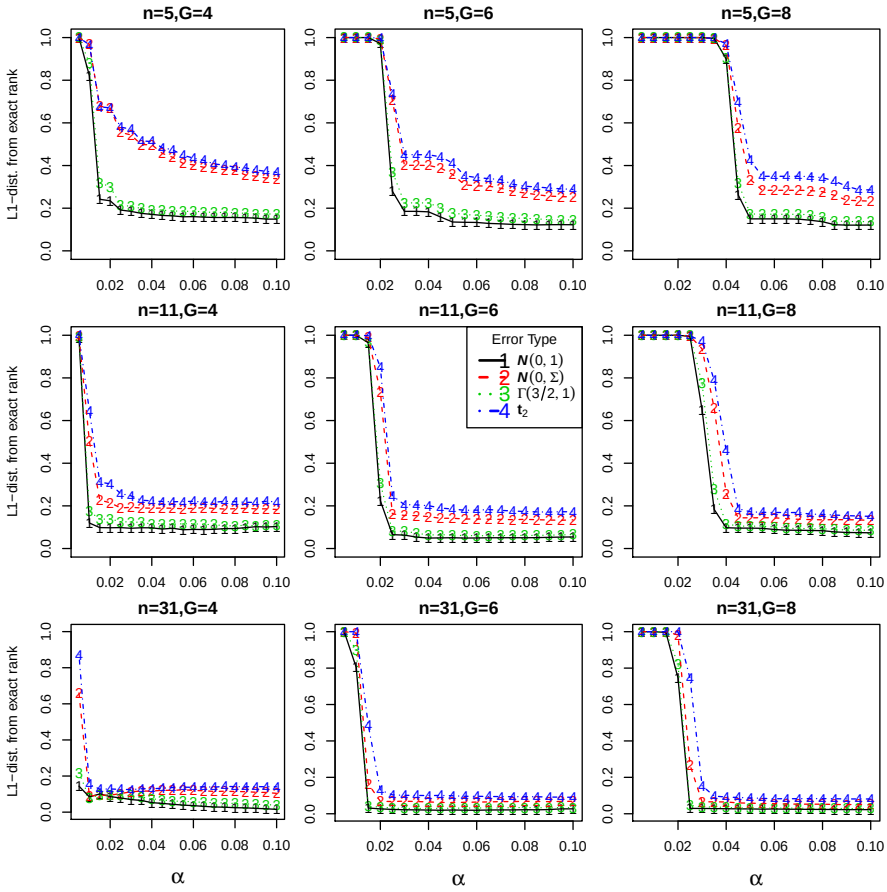

Figure 6: d_1 distance from true ranking, averaged across simulations, of the **"mean difference"** test statistic. The same graph for the **"t-test"** statistic is omitted as it is basically identical. The value of 1 represent the maximum distance from true ranking, hence the case in which all groups get the rank 1; 0 instead corresponds to perfect agreement between estimated and expected ranking.

In addition, an important note needs to be mentioned: for the "mean difference" and for "t-test" statistics the graphs shown are obtained with 7 variables, whereas for Hotelling's test statistic they are obtained with 37 variables. This because we have seen that under the null, the latter statistic has overfitting problems with $D = 7$ and $D = 13$ variables hence it is not reliable.

Here in the following are reported first the graphs related to the average d_1 distance (average across simulations), and then the graphs showing the proportion of exact ranking across simulations.

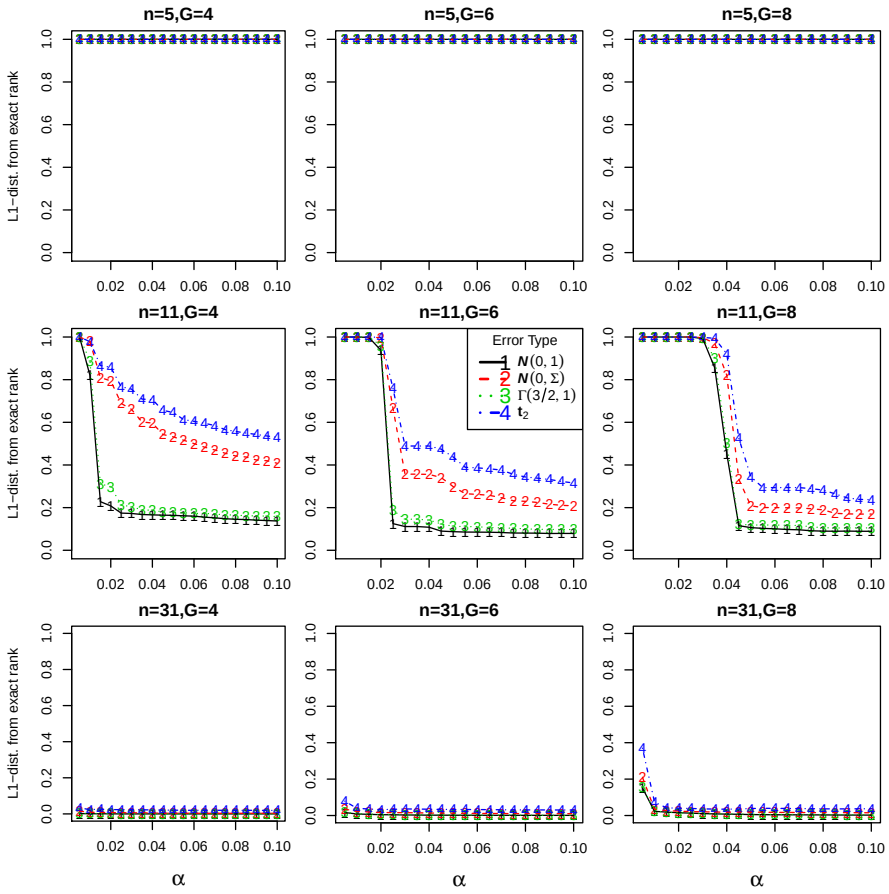

Figure 7: d_1 distance from true ranking, averaged across simulations, of the **Hotelling's** test statistic. Note that the first row of panels is constant as the test statistic was not computed in that setting combination because non existence of reference distribution for that settings combination.

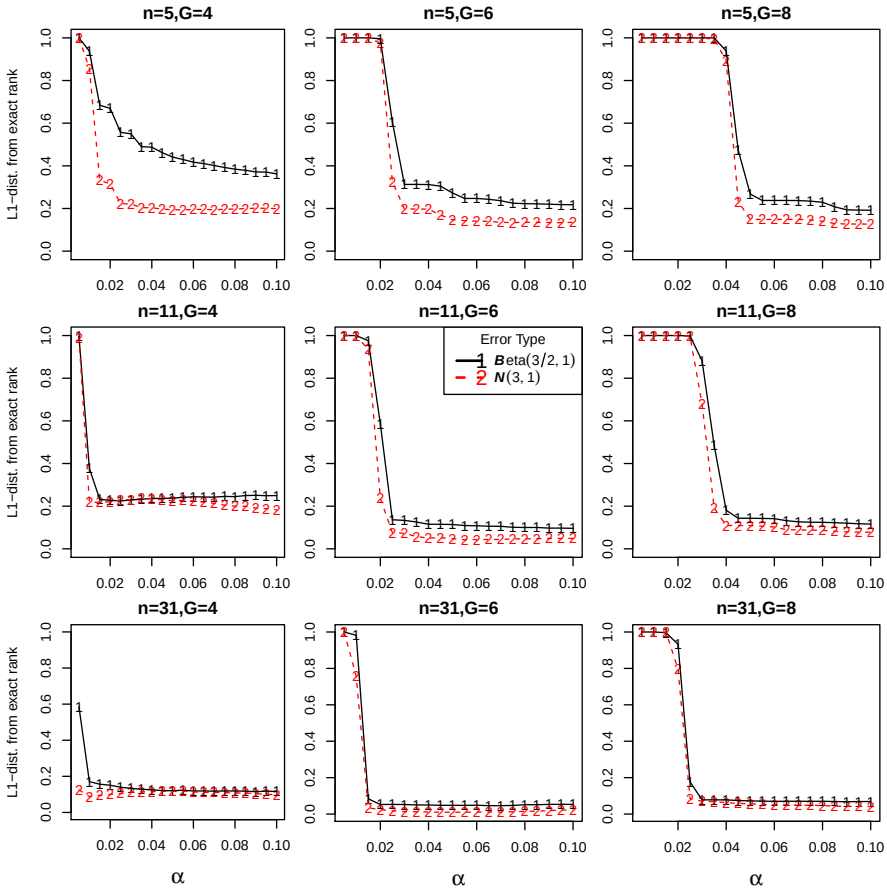

Figure 8: d_1 distance from true ranking, averaged across simulations, of the **"Anderson-Darling"** test statistic. The value of 1 represent the maximum distance from true ranking, hence the case in which all groups get the rank 1; 0 instead corresponds to perfect agreement between estimated and expected ranking.

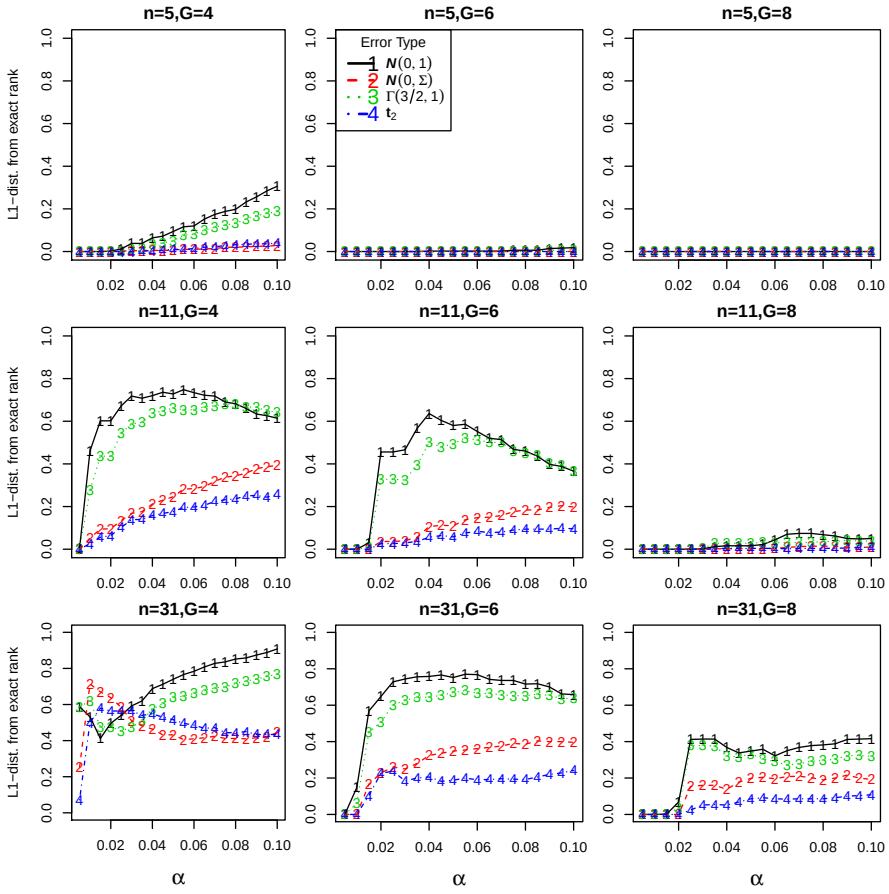

Figure 9: Proportion of "exact ranking" across simulations, *i.e.* $d_1 = 0$, for the **"mean difference"** test statistic. The same graph for the **"t-test"** statistic is omitted as it is basically identical.

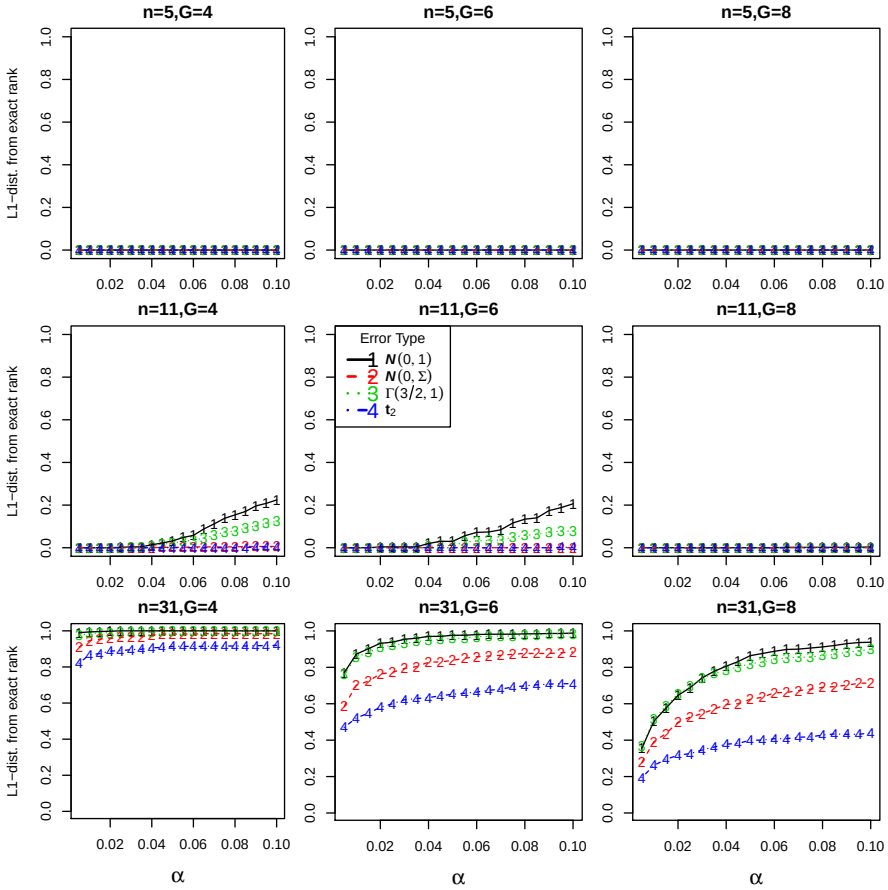

Figure 10: Proportion of "exact ranking" across simulations, *i.e.* $d_1 = 0$, for the **Hotelling's** test statistic. Here also the first row of panels is constant as the test statistic was not computed in that setting combination.

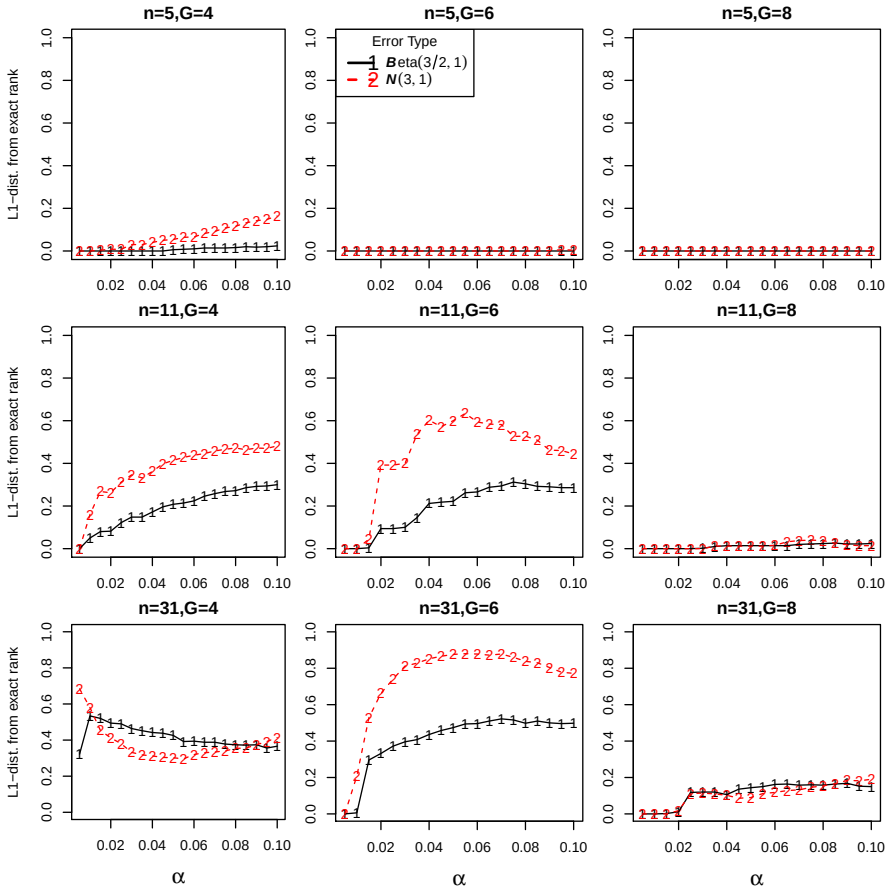

Figure 11: Proportion of "exact ranking" across simulations, *i.e.* $d_1 = 0$, for the **"Anderson-Darling"** test statistic.

From these graphs we can see that, in general, lines marked with "1" and "3"(black and light-green respectively) have a better behaviour as they are closer to zero in the "average" graph and higher than the others in the "exact ranking" graph. This is easily interpretable as the two lines correspond, respectively, to $\mathcal{N}(0,1)$ and $\Gamma(3/2,1)$ errors, hence the benchmark error distribution and the highly asymmetric and positive one. Indeed, in terms of information, the standard normal errors are carrying more information than the correlated normal ones whereas for the t_2 errors the large variance is responsible for the inferior performances. This is not entirely true for the Hotelling's test statistic as it gives better performances to the correlated normal errors *w.r.t.* the t_2 errors, this is due to the nature of the statistic that is specifically designed to exploit the correlation among variables.

Another general behaviour that is noticeable is that: increasing the number of groups makes it more difficult for the method to obtain an exact ranking, regardless of test statitstic and sample size; we can interpret that as an increased difficulty in trying to correctly estimate lot of parameters rather than few.

There is something else that sets the Hotelling's statistic apart in the "exact ranking" graph in the panels produced with sample size $n = 31$: although following the general behaviour, the proportion is always higher than the counterpart obtained with the other two statistics, regardless of the error type or the number of groups. This can be seen as an empirical confirmation of the difference in the speed with which the three test statistics diverge: for all of them increasing the sample size leads to more precise estimations, but for the Hotelling's statistic this increase is quicker, at least with these settings. Nonetheless, it still seems that "mean difference" and "t-test" are to be preferred over Hotelling's, given their better behaviour on lower sample sizes, unless indeed, the sample size is much larger than the number of variables.

Third Scenarios

In this scenario only the "exact ranking" graphs are reported as they are more informative than the "average graphs" in this situation and they basically convey the same information.

Although only 3 variables and 2 groups are active (active is intended here as *not under the null*), it is apparently enough for the algorithm to distinguish the right groups, at least for certain significance levels.

The clear increase in the proportion of exact ranking values marks when the active groups start to be clustered separately from the others and it is quite clear in the following graphs. This feature basically explain the behaviour of all curves in the two graphs that follows, the different shape and position of the increase mark the point from which the algorithm was able to correctly classify the active groups.

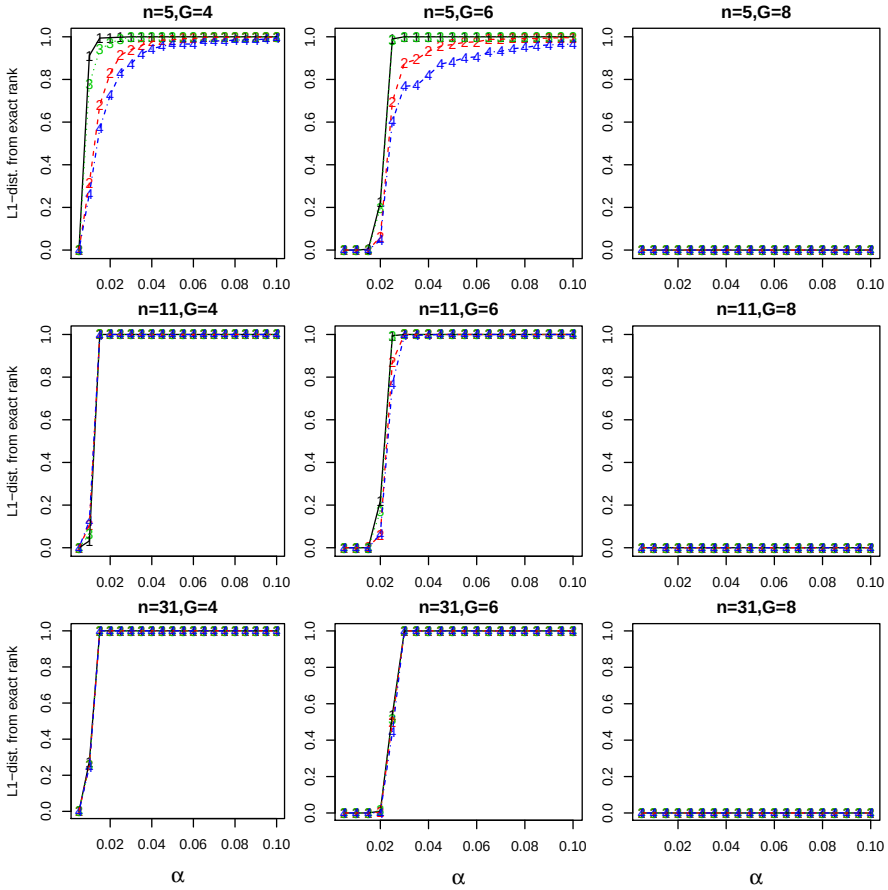

Figure 12: Proportion of "exact ranking" across simulations, *i.e.* $d_1 = 0$ for the **"mean difference"** test statistic. The same graph for the **"t-test"** statistic is omitted as it is basically identical.

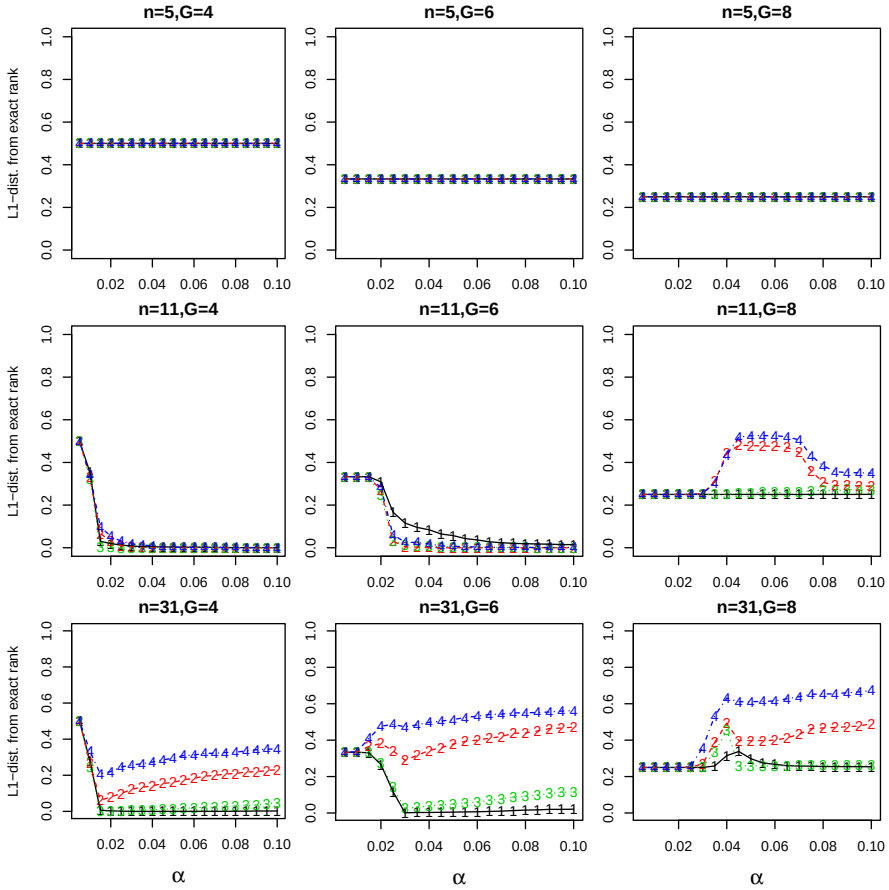

Figure 13: Proportion of "exact ranking" across simulations, *i.e.* $d_1 = 0$ for the **Hotelling's** test statistic. Here also the first row of panels is constant as the test statistic was not computed in that setting combination.

5 CASE STUDIES

In this chapter we will describe an applications of the method on a real dataset within the field of Statistical Genetics, it deals with microarray measurements for the lead optimisation stage in an early-stage drug development project.

5.1 GENE EXPRESSION STUDY

5.1.1 Some Background

A DNA microarray (also commonly known as gene chip, DNA chip, or biochip) is a collection of microscopic DNA spots attached to a solid surface. Scientists use DNA microarrays to measure the expression levels of large numbers of genes simultaneously or to genotype multiple regions of a genome. Each DNA spot contains picomoles (10^{-12} moles) of a specific DNA sequence, known as probes (or reporters). These can be a short section of a gene or other DNA element that are used to hybridize a cDNA or cRNA sample (called target) under high-stringency conditions. Probe-target hybridization is usually detected and quantified by detection of fluorophore-labeled targets to determine relative abundance of nucleic acid sequences in the target (wikipedia/microarray). A visualisation of a microarray experiment is depicted in-Figure14.

Studying gene expression (GE) with the help of microarray technology is highly interdisciplinary. It lies at a busy intersection of many different research areas. Microarray studies require input from molecular biology, bioinformatics and (bio)-statistics to design, carry out and interpret the results of these experiments. The current state of the technology would neither

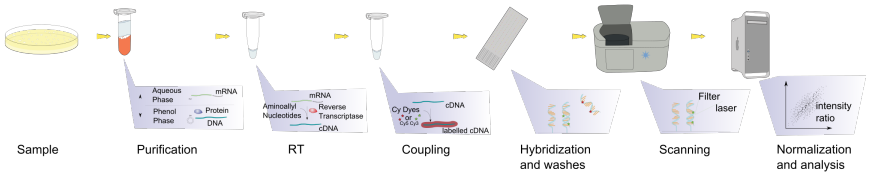

Figure 14: Steps required in a microarray experiment

have been possible without the advances in information technology, combinatorial chemistry and photo-lithography. But also physics has played a role in solving the "riddle of the bright mismatches" or attempts to define the boundaries for using the technology for absolute mRNA quantification (see Talloen and Göhlmann, 2009, for a comprehensive monography on the topic)

For these microarray data, pre-processing steps attempt to remove the technical variation originating from different sources during the process, from manufacturing of the microarrays to the biological and the microarray experiment. Methods used to remove the technical sources of variation, for background correction and normalization are performed as described in Talloen and Göhlmann (2009).

Next, genes are filtered with using the I/NI-call filtering (Talloen et al., 2007). The resulting data frame consists of a data matrix composed by positive values on the log scale measuring the expression of each gene (columns) in each biological sample (rows).

In this particular project, we are at the stage of lead optimisation: a step in the early drug development where chemists have synthesised typically 30 to 90 molecules (compounds) belonging to a limited number of chemotypes (class of similar compounds), and hence showing often only small variations in some of their substructures.

Moreover, a specifically developed biological experiment, or bioassay, is used to measure the effect (potency) of the compounds on the target identified for the disease under study. Other bioassays maybe used to measure off-target effects, which

are predictive for side-effects of the compound when consumed by humans.

The main aim for the chemists, at this stage, is to select which class of compounds, which chemotype, is best suited to be carried forward in the drug development process. One possibly useful indication, that can help in this choice, is to find which chemotype generates the least overall gene expression; indeed, in case such a chemotype is associated with a high bio-activity, it can be the most worthy to be further investigated because it induces the desired final biological effects while not interfering too much with the gene expression mechanism.

5.1.2 Analysis With SOUP

Hereafter we present the analysis made with the `SOUP` package and show the typical output of the `R` function. The package implements the method described in Chapter 3 on page 27 and it is available on CRAN.

The complete (already filtered) data matrix contains log-fold changes induced by 96 compounds on 7103 genes. The compounds are divided in 5 chemotypes. All these compounds were synthesised in order to target a specific receptor in the cell, the activation of which is related to the development of certain types of cancer in humans.

The sample size of groups are reported in Table 1.

Table 1: Sample sizes of chemotypes, considered as "treatments" in this study.

Chemotype label	2	3	5	6	8
	27	9	38	6	16

Given that we are looking for the chemotype generating the least overall gene expression (GE) we will have that, for each gene (variable), the lower the value the better; this can be set with the option "`tails`" in the call of the main function "`SOUP`". Moreover, as the original data matrix is too big to be handled by the algorithm, only a subset of all the 7103 genes was con-

sidered. This subset is the union of the 200 genes with the highest overall variance across compounds, and the genes already known from previous studies to be linked to the problem at hand.

Hereafter we report the estimated ranking as well as the output of the R function. In this analysis 2000 permutations were used. As test statistic we used the "t-test" because it seems a more sensitive choice, given the high unbalance of the group sample sizes and hence the possibly great difference in the variance of any two given groups. Indeed, studentised test statistics are preferable for permutation testing in non-ideal conditions such the present one: not only from the intuition point of view, but also from theoretical consideration about asymptotic robustness that can be found in the insightful paper of Chung and Romano (2013).

Thus the "asymptotic" p-values were employed and we also made use of the iterated NPC to avoid the additional burden of chosing the combining function, especially since we are trying to assume as less as possible in this exploratory study.

Table 2: Estimated ranking for the GE dataset.

Chemotype label	2	3	5	6	8
alpha=0.01	2	1	2	2	2
alpha=0.02	2	1	2	2	2
alpha=0.03	5	1	2	3	3
alpha=0.04	2	1	2	4	5
alpha=0.05	3	1	2	3	5
alpha=0.06	3	1	2	3	5
alpha=0.07	3	1	2	3	5
alpha=0.08	3	1	2	3	5
alpha=0.09	3	1	2	3	5
alpha=0.1	3	1	2	3	5
alpha=0.15	3	1	2	3	5
alpha=0.2	4	1	2	2	4

Hereafter we report the R output of the function, it summarises all important information.

```
*** "SoupObject" object of package "SOUP" ***
* Call:
```

```
SOUP(Y = geMatOK,  covars = as.integer(chemoTypes),  analysisType
     = "simple",
 p.adj.method = "FWEminP",  p.valuesType = "asymptotic",
 testStatistic = "Ttest",  univ.p.values = FALSE,
 tails = tails,  returnPermSpace = FALSE,  nPerms = 2000,  alpha
     = alpha,
 iteratedNPC = TRUE)


*** "RankResults" object of package "SOUP" ***
* Final ranking results *
           2 3 5 6 8
alpha=0.01 2 1 2 2 2
alpha=0.02 2 1 2 2 2
alpha=0.03 5 1 2 3 3
alpha=0.04 2 1 2 4 5
alpha=0.05 3 1 2 3 5
alpha=0.06 3 1 2 3 5
alpha=0.07 3 1 2 3 5
alpha=0.08 3 1 2 3 5
alpha=0.09 3 1 2 3 5
alpha=0.1  3 1 2 3 5
alpha=0.15 3 1 2 3 5
alpha=0.2  4 1 2 2 4

* Associated p.value matrix *
       2      3      5       6      8
2     NA 0.1710 0.0190 0.41779 0.0190
3 0.0030     NA 0.0025 0.00250 0.0025
5 0.0145 0.2635     NA 0.17100 0.0190
6 0.0940 0.3615 0.3615      NA 0.1710
8 0.0200 0.3370 0.1780 0.53373     NA

* p.values multiplicity adjustment method: FWEminP *
* Seed for the RNG:  565 *
*** End of "SoupObject" ***
```

From these results it seems that the chemotype number "3" produces the lowest GE so it would be the one to be chosen, but there is an important information missing: the bio-activity of those same compounds. Indeed, it can easily be that compounds of that chemotype have a low potency/activity, therefore they do not produce gene expression but they also lack

of biological activity and hence they are not interesting for the chemists.

This is something that clearly needs to be avoided, so in combination with these results it is reasonable to look at the bioactivity of the different chemotypes in comparison with their estimated rank. To this end we can, for example, plot the activity level for the two most important bioassays (that measures the potency on the target of interest) *vs.* their estimated rank to check which compounds are at the same time potent and with a good (low) rank.

The plot being described is reported in Figure 15 on the facing page, from the graph we can see that, although the chemotype "3" was selected as the one producing the least GE, it is also not the most potent and hence it may be not the most interesting from the biological point of view. Chemotype "5" instead, gets either rank two or three but it seems to be the most potent on the considered bioassays while still being not classified as the one producing the most GE.

Incidentally, and mostly because of biological considerations, the one being carried forward in the development process was chemotype "2", the one that seems the least potent. From our analysis it is classified as equal to chemotype "6", but apparently with the former one it was easier for the chemists to further improve its potency.

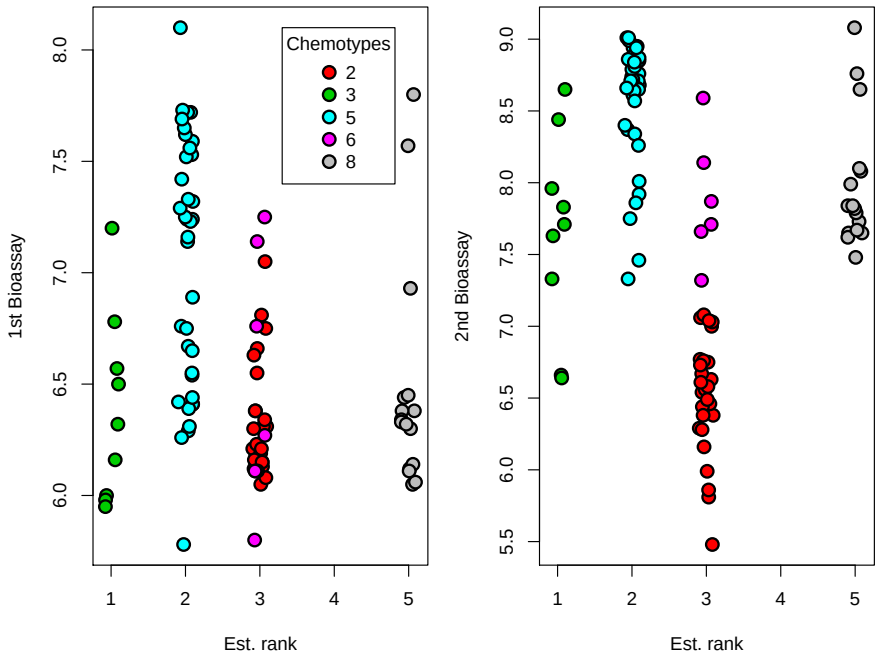

Figure 15: Values of the 2 most important bioassays plotted against the estimated ranking for each compound, the different colours represent the chemotype.

6 CONCLUSIONS

The aim of this thesis was developing a method capable of estimating the stochastic order of a set of multivariate treatments, by considering the joint information from more than one response variable.

By empirically comparing several experimental conditions, with 4 different test statistics, and by applying the method on a real dataset, we have demonstrated both the flexibility and the practical effectiveness of the algorithm constituting the main contribution of this thesis work.

The test statistics being employed can be tailored to the specific problem and data at hand, *e.g.* continuous or ordered categorical, in the presence of a stratifying variable or not, and can deal with several kind of errors.

The results of the simulations have also empirically demonstrated some features of the *NPC* (NonParametric Combination) in comparison with the *Hotelling's way* of combining information from more variables: the former seems to be more reliable and applicable to more cases, regardless of the sample size or dimensionality of the data, whereas the latter seems better if the sample size is considerably large but fails whenever the sample size is small and/or the errror variance is high (at least fails more often than the former).

This higher flexibility and reliability is clearly due to how the NPC is constructed. Since it considers one variable at-a-time, it can be used *e.g.* even when the number of variables is larger than the sample size. It thus summarises the information in an effective way. Furthermore, it permits the statistician (with a good knowledge of R) to write his own test statistic.

6.1 FUTURE RESEARCH

As far as future research is concerned, some directions in which more investigation could be of interest are the following.

6.1.1 Error Distribution

So far the error distributions considered in this work have different shape, correlation structure, and variance. It would be also interesting to assess the effect of heteroscedastic errors, *i.e.* error variance different from group to group, and correlated observations.

In the "heteroscedastic errors" situation, it would be useful to test the amount of variance needed *e.g.* in a single group for the method to breakdown, or even when the data are unbalanced as in that case the within-variance of the groups will be different by construction (unless pathological cases).

In the "correlated observation" situation, it could be interesting, for example, in case of multivariate time series data in which observations may have some sort of auto-correlation structure. The magnitude of this auto-correlation can be possibly tested by dividing the data groups with equal number of observations (respecting their order) and considering them as different "treatments". For example, within the *Statistical Process Control* framework, applying such procedure would be interesting in the sense that, if the global null hypothesis is rejected, it would be possible to pinpoint the moment when the system has started to deviate from the optimum in the multivariate sense.

6.1.2 Continuous Covariates

The current method is not capable of dealing with continuous covariates but only with one discrete covariate and possibly without a lot of categories. This because the permutation space

is restricted within each stratum and hence lowering the power in case the covariate has several levels.

A possibly more general approach could be for example to remove the linear effects of the covariates by pre-multiplying the data matrix by the square root of the matrix $I_n - H_x$ where H_x is the hat matrix coming from the linear model having as regressors the covariates. This would lead to an approximate solution in case of permutations but can still be a viable option for some practical experimental problems.

Another option would be to lose the distribution-free assumption typical of the permutation tests and to use the rotation tests instead. These tests assume left sphericity of the errors but are exact in case the assumption holds (see Solari et al., 2014, for details).

BIBLIOGRAPHY

Anderson, T. W. and Darling, D. A. (1952). Asymptotic theory of certain "goodness of fit" criteria based on stochastic processes. *Annals of Mathematical Statistics*, 23:193–212. (Cited on page 36.)

Anderson, T. W. and Darling, D. A. (1954). A test of goodness-of-fit. *Journal of the American Statistical Association*, 49:765–769. (Cited on page 36.)

Arboretti Giancristofaro, R., Corain, L., Gomiero, D., and Mattiello, F. (2010). Parametric *vs.* nonparametric approach for interval estimators of multivariate ranking parameters. Section on Nonparametric Statistics, Vancouver, British Columbia, Canada. *Joint Statistical Meeting*, *American Statistical Association*. (Cited on page 4.)

Basso, D., Pesarin, F., Salmaso, L., and Solari, A. (2009). *Permutation Tests for Stochastic Ordering and ANOVA*, volume Vol. 194 of *Lecture Notes in Statistics*. Springer. (Cited on pages 5 and 29.)

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300. (Cited on page 42.)

Benjamini, Y. and Yekutieli, D. (2001). The control of the false discovery rate in multiple testing under dependency. *Annals of Statistics*, 29(4):1165–1188. (Cited on page 42.)

Bringhurst, R. (1992). *The Elements of Typographic Style*. Hartley & Marks, Point Roberts, Washington, USA. (Cited on page 103.)

Chung, E. and Romano, J. P. (2013). Exact and asymptotically robust permutation tests. *The Annals of Statistics*, 41(2):484–507. (Cited on page 74.)

Finos, L., Pesarin, F., and Salmaso, L. (2003). Combined tests for controlling multiplicity in closed testing procedures (in italian). *Statistica Applicata*, 15(2):301–329. (Cited on page 41.)

Finos, L., Pesarin, F., Salmaso, L., and Solari, A. (2008). Exact inference for multivariate ordered alternatives. *Statistical Methods and Applications*, 17(2):195–208. (Cited on page 5.)

Finos, L., Salmaso, L., and Solari, A. (2007). Conditional inference under simultaneous stochastic ordering constraints. *Journal of Statistical Planning and Inference*, 137:2633–2641. (Cited on page 5.)

Gomiero, D. (2010). Parametric and non parametric mancova methods for comparing and ranking of multivariate populations,with application to industial experiments. Master's thesis, University of Padua, Faculty of Statistical Sciences. Supervisor prof. Fortunato Pesarin. (Cited on pages 34 and 36.)

Gupta, S. S. and Panchapakesan, S. (2002). *Multiple Decision Procedures*. Classics In Applied Mathematics. Siam, New York, 2nd edition. (Cited on page 4.)

Langsrud, Ø. (2005). Rotation tests. *Statistics and Computing*, (15):53–60. (Cited on page 8.)

Lehmann, E. L. and Romano, J. (2005). *Testing Statistical Hypotheses*. Springer, New York, 3rd edition. (Cited on page 11.)

Mattiello, F. (2010). Some resampling-based procedures for ranking of multivariate populations. Master's thesis, University of Padua, Faculty of Statistical Sciences. Supervisor prof. Fortunato Pesarin. (Cited on page 4.)

Mattiello, F., Bolzan, M., and Ravarotto, L. (2012). Assessing perceived food chemical risk by means of an educational project analysed with permutation tests. *Quaderni di Statistica*, 14:165–168. (Cited on page 4.)

Miede, A. (2010). *A Classic Thesis style*. http://www.ctan.org/tex-archive/macros/latex/contrib/classicthesis/ClassicThesis.pdf. (Cited on page 103.)

Pesarin, F. and Salmaso, L. (2010a). Finite-sample consistency of combination-based permutation tests with application to repeated measures designs. *Journal of Nonparametric Statistics*, 22(5):669–684. (Cited on page 2.)

Pesarin, F. and Salmaso, L. (2010b). *Permutation Tests for Complex Data*. Series in Probability and Statistics. Wiley & Sons, Chichester, U. K. (Cited on pages 1, 9, 18, 22, 23, and 41.)

Pettit, A. N. (1976). A two-sample anderson-darling rank statistic. *Biometrika*, 49:765–769. (Cited on page 37.)

Scheffé, H. (1943a). On a measure problem arising in the theory of non-parametric tests. *Annals of Mathematical Statistics*, 14:227–223. (Cited on page 13.)

Scheffé, H. (1943b). Statistical inference in the non-parametric case. *Annals of Mathematical Statistics*, 14:305–332. (Cited on page 13.)

Scholz, F. W. and Stephens, M. A. (1987). K-sample anderson-darling tests. *Journal of the American Statistical Association*, 82(399):918–924. (Cited on page 37.)

Shaffer, J. P. (1986). Modified sequentially rejective multiple test procedures. *Journal of the American Statistical Association*, 81(395):826–831. (Cited on page 41.)

Shaked, M. and Shanthikumar, J. G. (2007). *Stochastic Orders*. Springer Series in Statistics. Springer. 473 p. (Cited on pages 4 and 28.)

Solari, A., Finos, L., and Goeman, J. J. (2014). Rotation-based multiple testing in the multivariate linear model. *Biometrics*. (Cited on page 81.)

Talloen, W., Clevert, D.-A., Hochreiter, S., Amaratunga, D., Bijnens, L., Kass, S., and Göhlmann, H. W. (2007). I/ni-calls for the exclusion of non-informative genes: a highly effective filtering tool for microarray data. *Bioinformatics*, 23(21):2897–2902. (Cited on page 72.)

Talloen, W. and Göhlmann, H. (2009). *Gene Expression Studies Using Affymetrix Microarrays*. Mathematical & Computational Biology. Chapman and Hall/CRC. (Cited on page 72.)

Westfall, P. H. and Young, S. S. (1993). *Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment*. John Wiley, New York. (Cited on page 41.)

R CODE

In this chapter we report some of the R code implementing the method described in this thesis work, in particular:

1. the description file,
2. the script of the main function SOUP,
3. the function script computing the "mean difference" test statistic,
4. one of the most important utility functions, the one generating the matrix that pre-multiplied by the dataset gives the pairwise differences of means.

1 DESCRIPTION FILE

```
Package: SOUP
Type: Package
Title: Stochastic Ordering Using Permutations (and Pairwise Comparisons
    )
Version: 1.1
Date: 2011-11-16
Author: Federico Mattiello
Maintainer: Federico Mattiello <federico.mattiello@gmail.com>
Depends:
    R (>= 2.10.0),
    methods,
    tensor
Suggets: flip
Description: This package allows to construct a ranking of a set of
    treatments/groups, gathering together information coming from a
    several response variables.
    It can be used with both balanced and unbalanced experiments
    (with almost all test statistics) as well as in presence of either
    continuous covariates or a stratifying (categorical) variable.
License: GPL (>= 2)
LazyLoad: yes
Collate:
    'NPC.R'
    'PermSpace.R'
    'RankResults.R'
    'SOUP.R'
    'multiplicity.R'
    'simpleAD.R'
    'simpleHotelling.R'
    'simpleMeanDiff.R'
```

```
    'simpleTtest.R'
    'strataAD.R'
    'strataLmCoef.R'
    'strataMeanDiff.R'
    'strataTtest.R'
    't2p.R'
    'utilities.R'
    'PValueMat.R'
    'SoupObject.R'
    'rankingRule.R'
    'iterNPC.r'
```

2 MAIN FUNCTION

```
#' Main function of the package, interface for every analysis.
#' The dataset can be balanced or not for almost all possible choices
    of the
#' input parameters. The function allows also for the presence of one
    or more
#' continuous covariates or for stratified analysis.
#'
#' Depending on the chosen p-values type and on the analysis type, only
     some
#' options can be selected:
#' \itemize{
#' \item{}{
#'     with \code{"simple"} or \code{"regres"} analysis and
#'     \code{"asymptotic"} \emph{p}-values, \code{"Hotelling"} and
#'     \code{"Ttest"}; with \code{permutation} \emph{p}-values \code{"
    AD"},
#'     \code{"Hotelling"} and \code{"meanDiff"} can be selected.}
#' \item{}{
#'     With \code{"strata"} analysis and \code{"asymptotic"} \emph{p}-
    values,
#'     \code{"lmCoef"} and \code{"Ttest"}; with \code{"permutation"}
#'     \emph{p}-values \code{"AD"} and \code{"meanDiff"} can be
    selected.}
#' }
#' @title SOUP Main Function
#' @param Y
#'     input \code{matrix} where each column is a response variables.
#' @param covars
#'     it can be a \code{matrix}, a \code{data.frame} or a \code{
    formula},
#'     in the first two cases it must contains at least the labels of
    groups,
#'     in the latter case it has to be a right-sided \code{formula}
#'     (\emph{e.g.} \code{~ v1 + v2}) specifying the model to extract
    from
#'     the \code{data} input.
#' @param data
#'     optional \code{data.frame} containing covariates requested by \
    code{covars},
#'     if \code{covars} is not a formula this input is useless.
#' @param analysisType
#'     \code{character}, type of the analysis to be performed: it can
    be
#'     \code{"simple"} if the only covariate is the labels of groups,
```

```
#'     \code{"strata"} if there is also a stratifying (categorical)
    covariate,
#'     \code{"regres"} if there is one or more (numerical or not)
    covariate(s)
#'     besides labels of groups. In the latter case the linear effect
    of the
#'     covariates is removed from the response variables are
#'     residualised by the matrix \eqn{V^{-1/2}} obtained from
#'     \eqn{V = I - H} (where \eqn{I} is the identity matrix and \eqn{H
    } is
#'     the ''hat'' matrix of the OLS, by means of a spectral
    decomposition.
#' @param p.adj.method
#'     \code{character} string containing the type of required \emph{p
    }-value
#'     adjustment
#' @param p.valuesType
#'     \code{character} string indicating the type of \emph{p}-value to
     be
#'     used, it can be \code{"permutation"} or \code{"asymptotic"}
#' @param testStatistic
#'     \code{character} string indicating the test statistic to be used
    , it
#'     depends on both \code{analysisType} and on \code{p.valuesType}
    and
#'     the alternatives are:
#'     \describe{
#'         \item{\code{AD, meanDiff}}{
#'             for all \code{analysisType} but only using \code{
    permutation}
#'             \emph{p}-values}
#'         \item{\code{Ttest}}{
#'             for all \code{analysisType} but only using \code{
    asymptotic}
#'             \emph{p}-values}
#'         \item{\code{Hotelling}}{
#'             with both \code{permutation} and \code{asymptotic}
#'             emph{p}-values, with \code{"simple"} and \code{"regres"}
     but
#'             not with \code{"strata"} \code{analysisType}}
#'         \item{\code{lmCoef}}{
#'             only with \code{"strata"} \code{analysisType} and with
#'             \code{"asymptotic"} \emph{p}-values}
#'     }
#' @param combFunct
#'     \code{character} string containing the desired combining
    function to be
#'     used, choices are:
#'     \describe{
#'     \item{\code{Fisher}}{
#'         the famous Fisher's \emph{p}-values combining function}
#'     \item{\code{Liptak}}{
#'         it uses the quantile function of the Normal distribution to
    combine
#'         \emph{p}-values}
#'     \item{\code{minP, tippett}}{
#'         combine \emph{p}-values by taking the minimum across the set
    }
#'     \item{\code{maxT}}{
```

```
#'         combines directly the test statistics by taking the maximum
    across
#'         the set}
#'     \item{\code{direct, sumT}}{
#'         combine the test statistics by summing them}
#'     \item{\code{sumT2}}{
#'         combines the test statistics by squaring and summing them}
#'     }
#'     See the references for more details about their properties.
#' @param univ.p.values
#'     \code{logical}, if \code{TRUE} (default) \emph{p}-values are
    returned
#'     for each variable separately in a 3-ways \code{array}, the
    chosen
#'     multiplicity correction is performed independently for each
    variable
#' @param tails
#'     \code{integer} vector of \eqn{\pm 1}{\{+1,-1\}} containing the
#'     alternatives for response variables: \code{+1} means ‘‘the
    higher the
#'     better’’, \code{-1} means ‘‘the lower the better’’ (direction of
#'     preference), if \code{NULL} (default) all variables are
    considered
#'     to be of the type ‘‘the higher the better’’
#' @param linearInter
#'     \code{logical}, if \code{TRUE} the presence of linear
    interaction is
#'     assumed between levels of the stratifying covariate and response
#'     variables, this affects only the \code{"lmCoef"} test statistic
    in the
#'     (in the \code{"strata"} \code{analysisType}),
#'     basically the contrasts matrix of groups is multiplied by the
    levels
#'     of the stratifying factor.
#' @param returnPermSpace
#'     \code{logical} if \code{TRUE} (default) the whole permutation
    space is
#'     returned, class \code{\linkS4class{PermSpace}}, otherwise it is
    an empty
#'     instance of the class.
#' @param nPerms
#'     \code{integer} number of permutation to be performed
#' @param alpha
#'     \code{numeric} desired significance level, \emph{i.e.} type-I
    error
#' @param seed
#'     \code{integer} seed for the Random Number Generator
#' @param iteratedNPC
#'     \code{logical}, single or iterated Non-Parametric Combination,
    see \
#'     code{\link{iterNPC}} for details.
#' @param ...
#'     put here the optional \code{weights} and \code{subsets} for the
#'     \code{\link{NPC}} function and the permutation space of rows
    indexes
#'     \code{permSpaceID}.
#'     The latter allows to exactly reproduce a previous analysis, if
    all
#'     other inputs are kept equal, or to see what happens changing for
```

```
#'     example only the \code{testStatistic}.
#' @return
#'    an object of class \code{\linkS4class{SoupObject}}.
#' @author Federico Mattiello <federico.mattiello@@gmail.com>
#' @references
#'     Pesarin, F. and Salmaso, L. (2010)
#'     \emph{Permutation Tests for Complex Data}.
#'     Wiley: United Kingdom.\cr
#'
#'     Pesarin F. (2001)
#'     \emph{Multivariate Permutation Tests with Applications in
    Biostatistics}
#'     Wiley: New York.\cr
#'
#'     Federico Mattiello (2010)
#'     \emph{Some resampling-based procedures for ranking of
    multivariate
#'     populations}, Master's Thesis, Faculty of Statistical Sciences:
    Padova.
#'
#' @export
#' @examples
#' ###
#' ### testing SOUP
#' ###
#' rm(list = ls()); gc(reset = TRUE)
#'
#' require(SOUP)
#' n <- 5L         # replication of the experiment
#' G <- 4L         # number of groups
#' nVar <- 10L     # number of variables
#' shift <- 1.5    # shift to be added to group 3
#' alpha <- c(0.01, 0.05, 0.1)        # significance levels
#'
#' ## groups factor
#' groups <- gl(G, n, labels = paste("gr", seq_len(n), sep = "_"))
#'
#' set.seed(12345)
#' Y <- matrix(rnorm(n * G * nVar), nrow = n * G, ncol = nVar)
#' colnames(Y) <- paste("var", seq_len(nVar), sep = "_")
#' ind1 <- groups == unique(groups)[3L]
#' Y[ind1, ] <- Y[ind1, ] + shift
#'
#' res <- SOUP(Y = Y, covars = as.matrix(groups), analysisType = "
    simple",
#'         testStatistic = "meanDiff", combFunct = "Fisher",
#'         alpha = alpha,
#'         subsets = list("first" = 1:5, "second" = 6:10),
#'         weights = list(
#'                 "firstW" = c(.1, .2, .1, .5, .1),
#'                 "secondW" = rep.int(1, 5)),
#'         p.valuesType = "permutation", p.adj.method = "FWEminP")
#' res
#'
SOUP <- function(Y, covars, data = NULL, analysisType, p.adj.method, p.
    valuesType,
    testStatistic, combFunct, univ.p.values = TRUE, tails = NULL,
    linearInter = FALSE, returnPermSpace = TRUE,
    nPerms = 999L, alpha = 0.05, seed, iteratedNPC, ...)
```

```
{

    ###
    ### Matching arguments
    ###

    ### iteratedNPC, if missing is set to FALSE
    if (missing(iteratedNPC))
    {
        iteratedNPC <- FALSE
    } else {}

    ### analysisType
    if(missing(analysisType))
    {
        stop("\"analysisType\" is missing, must be one of ",
                "\'simple\', \'strata\' or \'regres\'"
        )# END:stop
    } else {
        analysisType <- match.arg(analysisType, c("simple", "strata", "
            regres"))
    }# END:if-analysisType


    ### covars, i.e. covariate(s)
    if(missing(covars) || is.null(covars))
    {
        stop("\"covars\" is empty. It must contains at least the ",
                "vector of groups\' labels")
    } else {}# END:covars-present


    ### p.adj.method
    if(missing(p.adj.method))
    {
        ## default = bonferroni
        warning ("p.value adjustment method is missing, ",
            "using \"BHS\" as default", call. = FALSE)
        p.adj.method <- "BHS"
    } else {
        p.adj.method <- match.arg(p.adj.method, c("BHS", "FWEminP", p.
            adjust.methods))
    }# END:if-p.adj.method


    ### p.value.type
    if(missing(p.valuesType))
    {
        stop("\"p.valuesType\" is missing, must be either ",
                "\"asymptotic\" or \"permutation\"")
    } else {
        p.valuesType <- match.arg(p.valuesType, c("asymptotic", "
            permutation"))
    }# END: missing - p.valuesType


    ### selecting the proper "testStatistic": depends on "analysisType"
    ### and "p.valuesType"
    if(missing(testStatistic))
```

```
{
    stop(
        "\"testStatistic\" must be one of: ",
        "\"AD\", \"lmCoef\", \"Hotelling\", \"meanDiff\" or \"Ttest
            \""
    )
} else
{
    if(p.valuesType == "asymptotic")
    {
        switch(analysisType,
            "simple" = {
                testStatistic <- match.arg(testStatistic,
                        c("Hotelling", "Ttest"))
            },
            "strata" = {
                testStatistic <- match.arg(testStatistic,
                        c("lmCoef", "Ttest"))
            },
            "regres" = {
                testStatistic <- match.arg(testStatistic,
                        c("Hotelling", "Ttest"))
            }
        )# END: switch - analysisType
    } else ##  permutation p.values
    {
        switch(analysisType,
            "simple" = {
                testStatistic <- match.arg(testStatistic,
                        c("AD", "Hotelling", "meanDiff"))
            },
            "strata" = {
                testStatistic <- match.arg(testStatistic,
                        c("AD", "meanDiff"))
            },
            "regres" = {
                testStatistic <- match.arg(testStatistic,
                        c("AD", "Hotelling", "meanDiff"))
            }
        )# END:switch-analysisType
    }# END:if-p.valuesType
}# END:missing-testStatistic


### checks for Hotelling's test-statistic
if(testStatistic == "Hotelling")
{
    if(univ.p.values)
    {
        warning ("When using \"Hotelling\" test-statistic ",
                "univariate p-values are not calculated.",
            call. = FALSE)# END:warning
        univ.p.values <- FALSE
    } else {}# END:if-univ.p.values

    if(analysisType != "simple")
    {
        warning ("When using \"Hotelling\" test-statistic no ",
                "covariates are considered, so only \"simple\" ",
```

```
                "analysis can be performed.", call. = FALSE)
        analysisType <- "simple"
    } else { }# END: if - simple
}# END: check - hotelling


### check combining function
if(missing(combFunct))
{
    if (testStatistic != "Hotelling")
    {
        message ("combining function is missing, ",
                "using \"Fisher\" as default")
        combFunct <- "fisher"
    } else {}
} else {
    combFunct <- match.arg(tolower(combFunct),
        c("fisher", "liptak", "minp", "tippett",
                "maxt", "sumt", "direct", "sumt2"))
}# END: missing - combFunct


### check of tails
if(missing(tails) || is.null(tails))
{
    tails <- rep.int(1L, NCOL(Y))
} else
{
    if(length(tails) != NCOL(Y))
    {
        warning ("Number of \"tails\" differs from number of ",
                "variables. \"tails\" are all set to 1.",
                call. = FALSE)
        tails <- rep.int(1L, NCOL(Y))
    } else {}
}# END: if - check tails


### extract arguments in '...'
dots <- list(...)


### check dim of Y
if(NCOL(Y) == 1L)
{
    dim(Y) <- c(NROW(Y), 1)
} else {}# END:dim-Y



### 'covars' is a formula of variables in 'data'
if(is(covars, "formula"))
{
    if(length(covars) > 2L)
    {
        warning("\"covars\" is a \"formula\" with response, ",
                "it has to be a right-sided formula ",
                "(responses are in \"Y\").")
            covars <- covars[-2]
```

```
    } else {}# END: if - formula check

    if(!is.null(data))
    {
        covars <- model.frame(covars, data = data)
    } else {
        stop("\"data\" argument is missing and \"covars\" is a
            formula, ",
                "not a matrix.")
    }# END: ifelse - data check
} else {}# END: ifelse - covars is a formula


### "covars" is a matrix
if(!is.data.frame(covars))
{
    covars <- data.frame(covars)
} else {}# END:if



### set the seed for the random number generator
if(missing(seed))
{
    seed <- round(1e3 * runif(1), digits = 0)
}# END:ifelse-set.seed
set.seed(seed)

###
### END Matching
###


### regression aov: residualise Y with a linear regression of Y on
    the
### covariate(s); after that is like the "simple"
if(analysisType == "regres")
{
    Xmm <- model.matrix(model.frame(covars[, -1L, drop = FALSE]),
        data = data)
    Y <- .orthoX(Y, Xmm)
    analysisType <- "simple"
    covars <- covars[, 1L, drop = FALSE]
} else {}# END: if - regres



### do the analysis
switch(analysisType,

    ### simple aov: AD = Anderson-Darling, Hotelling = Hotelling's
        T^2,
    ##  meanDiff = differences of means
    "simple" = {
        ### check number of columns of covars
        if(NCOL(covars) > 1L)
        {
            stop("selected \"simple\" so \"covars\" must have only
                one column")
```

```
        }# END:if-no-covariate(s)

        ### ordering of data
        groups <- covars[, 1]

        ord <- order(groups)
        Y <- Y[ord, ]
        groups <- groups[ord]

        ### permutation space of row IDs
        if(is.null(dots$permSpaceID))
        {
            permIndexMat <- .makePermSpaceID(nObs = NROW(Y),
                analysisType = analysisType, seed = seed, nPerms =
                    nPerms)
        } else
        {   ## take input permSpace of indexes (for reproducibility
            )
            permIndexMat <- dots$permSpaceID
            seed <- integer()
        }# END: indexes - permSpace

        switch(testStatistic,
            "AD"        = {
                T <- .simpleAD (
                    dataset = Y, groups = groups, indexMat =
                        permIndexMat
                )# END:AD
            },
            "Hotelling" = {
                T <- .simpleHotelling (
                    dataset = Y, groups = groups, indexMat =
                        permIndexMat,
                    p.valuesType = p.valuesType
                )# END:hotelling
            },
            "meanDiff"  = {
                T <- .simpleMeanDiff (
                    dataset = Y, groups = groups, indexMat =
                        permIndexMat
                )# END:meanDiff
            },
            "Ttest"  = {
                T <- .simpleTtest(
                    dataset = Y, groups = groups, indexMat =
                        permIndexMat
                )# END:Ttest
            },
            {
                stop("can not match the test statistics")
            }
        )# END:switch-simple
    },

    ### stratified aov: "AD" = Anderson-Darling, "lmCoef" = anova
        test
    ### for the effect of the "groups" and pairwise differences
        between
    ### coefficients estimated by "lm", variable by variable,
```

```
### "meanDiff" = differences of mean, "Ttest" = t-test as in
### the case "simple"-"meanDiff"-asymptotic p.values but with
### different denominator (the MSE take into account also the
### stratifying variable considered as factor)
"strata" = {
    ### check number of columns of covars
    if(NCOL(covars) > 2)
    {
        stop("selected \"strata\" so \"covars\" must have 2
            columns, ",
                "both factor variables")
    }# END:if-no-covariate(s)

    ### ordering of data
    # if((NCOL(covars) == 2) && is.factor(covars[, 1]) && is.
        factor(covars[, 2])) {
    if(NCOL(covars) == 2L)
    {
        ord    <- order(covars[, 2], covars[, 1])
        covars <- covars[ord, ]
        Y      <- Y[ord, ]
        groups <- covars[, 1]
        strata <- .unfactor(covars[, 2])
    }# END:if-1-covariate
    ### permutation space of rows IDs
    if(is.null(dots$permSpaceID)) {
        permIndexMat <- .makePermSpaceID(nObs = NROW(Y),
            analysisType = analysisType, strata = strata,
            seed = seed, nPerms = nPerms
        )
    } else {# take input permSpace of indexes (for
        reproducibility)
        permIndexMat <- dots$permSpaceID
        seed <- integer()
    }# END:indexes-permSpace
    ### select test statistic
    switch(testStatistic,
        "AD"       = {
            T <- .strataAD(
                dataset = Y, groups = groups, strata = strata,
                indexMat = permIndexMat
            )# END:AD
        },
        "lmCoef"    = {
            T <- .strataLmCoef(
                dataset = Y, groups = groups, strata = strata,
                indexMat = permIndexMat, linearInter =
                    linearInter
            )# END:lmCoef
        },
        "meanDiff" = {
            T <- .strataMeanDiff(
                dataset = Y, groups = groups, strata = strata,
                indexMat = permIndexMat, linearInter =
                    linearInter
            )# END:meanDiff
        },
        "Ttest"    = {
            T <- .strataTtest(
```

```
                    dataset = Y, groups = groups, strata = strata,
                    indexMat = permIndexMat, linearInter =
                        linearInter
                )# END:Ttest
            },
            {
                stop("can not match the test statistics")
            }
        )# END:switch-testStatistic-strata
    }
)# END:switch-analysisType
###----------------------------##


### from raw statistics to univariate p.values
if(p.valuesType == "asymptotic")
{
    P <- T
} else
{
    P <- t2p(T)
}# END:ifelse-asympotic-p.values

### create object PValueMat
if(univ.p.values)
{
    pValueMat <- .makePValueMat(
        P = P, multAdjMethod = p.adj.method, groupsLabs = unique(
            groups)
    )
} else
{
    pValueMat <- new("PValueMat")
}# END:if-univ.p.values


### NPC and create object "PermSpace"; subsets and weights are
### extracted from "..."
if(testStatistic == "Hotelling")
{
    T.H0Low <- T
    if(p.valuesType == "asymptotic")
    {
        T.H0Gre <- 1 - T.H0Low
    } else
    {
        T.H0Gre <- -T.H0Low
    }# END:asymptotic

    permSpace <- new(
        Class      = "PermSpace",
        seed       = seed,
        T.H0Low    = T.H0Low,
        T.H0Gre    = T.H0Gre,
        P.H0Low    = t2p(T.H0Low),
        P.H0Gre    = t2p(T.H0Gre),
        rawStats   = array(0, dim = c(0, 0, 0)),
        comb.funct = combFunct)
} else
```

```
    {
        ### change behaviour: if iteratedNPC then add permSpace2 to the
             output list
#        permSpace <- NPC (rawStats = T, combFun = combFunct, seed =
    seed,
#                p.values = (p.valuesType == "asymptotic"),
#                tails = tails, subsets = dots$subsets, weights = dots$
    weights,
#                iteratedNPC = FALSE)
        permSpace <- NPC (rawStats = T, combFun = combFunct, seed =
            seed,
                p.values = (p.valuesType == "asymptotic"),
                tails = tails, subsets = dots$subsets, weights = dots$
                    weights,
                iteratedNPC = iteratedNPC)
    }# END:if-permSpace-generation
    permSpace@IDs <- permIndexMat

    ### create object "RankResults"
    rankRes <- rankingRule(permSpace = permSpace, alpha = alpha,
            multAdjMethod = p.adj.method, groupsLabs = unique(groups))

    ### conditional removal of PermSpace
    if (!returnPermSpace)
    {
        permSpace <- new(Class = "PermSpace", seed = seed, rawStats = T
            )
    } else {}

    ### create object "SoupObject"
    soupRes <- new("SoupObject",
        call        = match.call(),
        rankResults = as.list(rankRes),
        pValueMat  = pValueMat,
        permSpace   = permSpace)

    return(soupRes)

}#=END=
```

3 MEAN DIFFERENCE SCRIPT

```
.simpleMeanDiff <- function(dataset, groups, indexMat)
{
    #- global variables
    groups <- as.factor(groups)
    B <- NCOL(indexMat)
    nObs <- NROW(dataset)
    p <- NCOL(dataset)
    C <- length(tab <- table(groups))
    K <- C * (C - 1)/2

    ##- labels
    labsMat <- t(outer(levels(groups), levels(groups), FUN = paste, sep
         = "-"))
    labsPC <- labsMat[lower.tri(labsMat)]

    #- matrix of statistics
```

```
    T <- array(NA, c(B + 1, p, K))

    #- Contrasts Matrix (CM), and checking (un)balance of the
        experiment
    CM <- .DesM(tab)/rep(tab, tab)

    #- observed statistics
    Ttemp <- t(dataset) %*% CM
    T[1, , ] <- Ttemp

    #- permutation statistics
    for(bb in 2L:(B + 1))
    {
        ind <- indexMat[, bb - 1]
        data.p <- dataset[ind, , drop = FALSE]
        Ttemp <- crossprod(data.p, CM)
        T[bb, , ] <- Ttemp
    }# END:for-bb
    ##- last permutation==observed
#    T[B + 1, , ] <- T[1, , ]

    dimnames(T) <- list(
        c("p-obs", paste("p-*", seq_len(B), sep = "")), colnames(
            dataset), labsPC
    )
    return(T)
}#=END=
```

4 PAIRWISE DIFFERENCE MATRIX

```
##============================================================##
##  Constructing Design Matrix for pairwise comparisons       ##
##------------------------------------------------------------##
## @author: Federico Mattiello                                ##
## @date:   17/10/2011                                        ##
## @version: 3.0                                              ##
## @notes: unbalanced case taken into account                 ##
## @notes: now 4 times faster thanks to the use of "matrix"   ##
##         instead of "kronecker"                             ##
##------------------------------------------------------------##
## Inputs:                                                    ##
## - N: number of replications of the experiment, either      ##
##      an integer vector or a "table", both of length "C"    ##
##------------------------------------------------------------##
## Outputs:                                                   ##
## - M: matrix of {-1, 0, 1}, of dimensions (sum(N) x K)      ##
##      where K = C*(C-1)/2; if dataset has "nObs" rows and   ##
##     "p" columns then "t(dataset) %*% M = P", where P is    ##
##      a (p x K) matrix of pairwise differences of sums      ##
##============================================================##
#' Construct Design Matrix that pre-multiplied to the dataset gives the
#' pairwise mean differences (wrt the groups)
#'
#' @title Design Matrix For Pairwise Differences
#' @rdname DesM
#' @param N
#'     number of replication of the experiment for each group, either
    an
```

```
#'     \code{integer} vector or a \code{table} of length \code{G}
#'     \emph{i.e.} number of groups/treatments
#' @return
#'     matrix of
#' @author Federico Mattiello <Federico.Mattiello@@UGent.be>
#'
.DesM <- function(N)
{
    M <- NULL
    C <- length(N)
    sq <- c(0, cumsum(N))
    for (i in seq_len(C - 1))
    {
        tmp <- NULL
        negD <- -diag(C - i)
        for (j in (i + 1):C)
        {
            tmp <- rbind(tmp,
                matrix(negD[j - i, ], nrow = N[j], ncol = C - i, byrow
                    = TRUE)
            )# END:tmp
        }# END:for-j
        A <- rbind(
            array(0, c(sq[i], C - i)),
            array(1, c(N[i], C - i)),
            tmp
        )# END:A
        M <- cbind(M, A)
    }# END:for-i
    return(M)
}#=END=
```

COLOPHON

This thesis work has been written with LaTeX2$_\varepsilon$, a free typesetting system, particularly useful to elaborate scientifical documents, and using a re-elaboration of the "ClassicThesis" style by André Miede (Miede, 2010). The ClassicThesis style, inspired by the work of Robert Bringhurst "The Elements of Typographic Style" (Bringhurst, 1992), is available on CTAN.

NOTE: The font family used in this work is font *Palatino* by Hermann Zapf. Mathematical formulas are composed with the Hermann Zapf and Donald Knuth's fonts *AMS Euler*. Fixed space font is *Bera Mono*, previously developed by Bitstream Inc. as "Bitstream Vera". Sans-serif style fonts are *Iwona*, by Janusz M. Nowacki.

Final version: October 31, 2014.